Technical Report 1001

Natural Object Categorization

Aaron F. Bobick

MIT Artificial Intelligence Laboratory

Natural Object Categorization

by

Aaron F. Bobick

Massachusetts Institute of Technology

November, 1987

Revised version of a thesis submitted to the Department of Brain and Cognitive Sciences on July 22, 1987 in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Natural Object Categorization

Aaron F. Bobick

Abstract

This thesis addresses the problem of categorizing natural objects. To provide a criteria for categorization we propose that the purpose of a categorization is to support the inference of unobserved properties of objects from the observed properties. Because no such set of categories can be constructed in an arbitrary world, we present the Principle of Natural Modes as a claim about the structure of the world.

We first define an evaluation function that measures how well a set of categories supports the inference goals of the observer. Entropy measures for *property uncertainty* and *category uncertainty* are combined through a free parameter that reflects the goals of the observer. Natural categorizations are shown to be those that are stable with respect to this free parameter. The evaluation function is tested in the domain of leaves and is found to be sensitive to the structure of the natural categories corresponding to the different species.

We next develop a categorization paradigm that utilizes the categorization evaluation function in recovering natural categories. A statistical hypothesis generation algorithm is presented that is shown to be an effective categorization procedure. Examples drawn from several natural domains are presented, including data known to be a difficult test case for numerical categorization techniques. We next extend the categorization paradigm such that multiple levels of natural categories are recovered; by means of recursively invoking the categorization procedure both the genera and species are recovered in a population of anaerobic bacteria.

Finally, a method is presented for evaluating the utility of features in recovering natural categories. This method also provides a mechanism for determining which features are constrained by the different processes present in a multiple modal world.

Thesis Supervisor: Dr. Whitman Richards
Professor, Department of Brain and Cognitive Sciences

Acknowledgments

Many, if not most, of the original ideas presented in this thesis are the result of the numerous discussions I have had with the faculty, staff, and graduate students at MIT. It has been my experience that the faculty in the Brain and Cognitive Sciences Dept. treat the graduate students as colleagues; such respect has been important in making the doctoral process a stimulating and rewarding learning experience. In particular, I thank Merrill Garrett and Jerry Fodor for introducing me to the problems of Cognitive Science. As an undergraduate in computer science, I learned from them the remarkable computing ability of the human brain. Molly Potter and Shimon Ullman have kept me honest, questioning assumptions and always asking critical questions.

The most important interaction I have had as a student at MIT has been with my advisor, Whitman Richards. Not only has he been a continual source of stimulating ideas, but he has also provided professional and moral support rarely found in the academic community.

I thank the additional members of my committee — Professors Patrick Winston, David Mumford, Shimon Ullman, and Steven Pinker — for their thoughtful comments and critiques.

As fellow graduate students Chris Atkeson (now *Professor* Atkeson!) and Eric Saund have provided both fascinating discussions — sometimes about exciting ideas, sometimes about absurd speculations — and moral support. Eric has the uncanny ability to ask me annoyingly simple yet important questions for which I have absolutely no good answer.

Finally, I thank my wife Denise, to whom this thesis is dedicated and whose continued love and support and most of all friendship have made this endeavor possible. I eagerly look forward to being able to return to my role of supportive husband and friend.

This thesis describes research done at the Department of Brain and Cognitive Sciences and the Artificial Intelligence Laboratory at the Massachusetts Institute of Technology. Support for this research is provided in part by the National Science Foundation grant ISI–8312240 and AFOSR grant F49620–83–C–0135. Support for the research at the Artificial Intelligence Laboratory is provided in part by the Systems Development Foundation and the Defense Advanced Research Projects Agency under Office of Naval Research contracts N00014–80–C–0505, N00014-82-K-0334, N00014-85-K-0124.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Problem: Object Categorization

Let us travel back to the jungle of our ancestors. We see an object in the distance, moving slowly on four legs. The object has black stripes on a beige coat of fur, a large appendage in front (the "head") with sharp serrations in a hinged opening, long whisker-like hairs in front, and a narrow, elongated rear appendage that oscillates. Suddenly, we notice the object has turned and two round, black objects, recessed in the front appendage, are now pointed in our direction. As it begins to move toward us, we quickly decide that this is an appropriate time to leave, and with due haste.

If analyzed only casually, the above scenario appears to be an example of simple and rational behavior. We view an object which we perceive to be a tiger, we know that tigers feast on people, and thus we decide to run for our lives. But let us examine the scenario in greater detail. Our first (perceptual) act is to encode some stimulus information: an object[1] with four downward pointing appendages, translating across our visual field, endowed with certain physical characteristics. Our last (behavioral) act is a decision to flee, based upon knowledge of the potential behavior of that object. But, somewhere in between those two events, we make the critical inference about unobserved properties of an object from the observed properties. Given only a sensory description of an object, we are able to make inferences about unobservable

[1] For this example, and in fact for the entire thesis, we ignore the question of how we know that some part of the visual stimulus comprised an "object," a single entity.

properties such as the intentions of an animal. How is such an inference possible?

The obvious, in fact seemingly trivial, answer is that sensory information available is sufficient to determine that the object is a tiger; thus, our knowledge about the behavior of tigers allows us to predict the behavior of the object. That is, given the sensory information, we conclude that the object is a member of the "tiger" *category* and thus we expect the object to behave in a manner consistent with the behavior of other objects of the same category.

But this answer to the question of how the observer makes predictions about the behavior of objects is not adequate. Simply announcing a category to which an object belongs does not provide the observer with the necessary predictive power. For example, suppose we view the previously described situation, but decide that the object in question belongs to the category "large fuzzy thing." In this case, our ability to make inferences about the behavior of the object is limited, and our response might not be appropriate for the situation. The large fuzzy thing would partake of an early supper. Although the category asserted is correct, "large fuzzy thing" does not support the inferences that are necessary for observer to interact successfully with his environment.

However, the intuition that the observer accomplishes his inference task by determining the "correct" category of an object is strong. The only difficulty with the previous example was that some categories (like "tiger") are more useful for inference than others ("large fuzzy thing"). Therefore, if the observer is to predict the important behavior of objects by determining the categories to which they belong, then those categories must be matched both to the goals of the observer and to the structure of the world. In particular, these categories must satisfy two requirements. First, using only sensory information, the observer must be able to determine the category to which an object belongs. Second, once the category of an object is established, membership of the object in that category must allow the observer to make important inferences about the behavior of the object. Which inferences are important depends upon the goals of the observer.

As we will discuss in the next section, we have no a priori reason to believe that a set of categories exist that permits the observer to both identify the category of an object from sensory information and predict unobserved properties as well. And if such categories do exist, how would the observer

come to know them? The goal of this thesis is to understand and provide a solution to the problem of discovering the useful categories in the world.

We can decompose the object categorization problem into the following three questions:

- What are the necessary conditions that must be true of the world if a set of categories is to be useful to the observer in predicting the important properties of objects?
- What are the characteristics of such a set of categories?
- How does the observer acquire the categories that support the inferences required?

These problems follow one another directly. By identifying the structure in the world that must be present in order for the observer to be able to construct a set of categories that supports important inferences, we are able to specify the characteristic structure that such a set of categories must exhibit. Once we have identified these characteristics we can attempt to recover categories that satisfy these conditions.

1.2 A Necessary Condition: Natural Modes

We have stated that goal of categorization is to permit the inference of important properties of objects. Often, however, many of the important properties of objects are not directly observable. There is no direct sensory stimulus for "tends to eat human beings for dinner." Thus, if the observer it to accomplish this categorization task, then he is required to predict unobserved properties from observed properties. How is this possible? Certainly, one could construct a world in which the inference task was not feasible. If the important (unobserved) properties of objects are independent of the properties available to the observer through his sensory mechanisms, then no useful inferences could be made. No set of categories could be constructed that would allow the observer to predict the behavior of objects. Therefore, if we assume that useful categorization is possible, if we accept human perception as an existence proof that the goal of making reliable inferences about the properties of objects can be achieved, then it must be the case that our world structured in a special way.

To capture this structuring of the world, we propose the Principle of Natural Modes, a claim that the world does not consist of arbitrary objects, but of objects highly constrained by the processes that create them and the environment that acts upon them. Natural modes — clusterings of objects in properties important to the interaction between objects and their environment — cause objects to display large degrees of redundancy; for example, most objects with beaks also have wings, claws, and feathers. Because objects within the same natural mode exhibit the same behavior in terms of their important properties, the natural modes are an appropriate sets of categories for the recognition task. Once the natural mode of an object is established, important properties of that object can be inferred. Stated succinctly, natural modes provide the basis for a natural categorization of the world.

The goal of the observer, then, is to recover the natural mode categories in the world. Our task is to develop the theoretical tools necessary to allow the observer to accomplish achieve his goal. In the chapters that follow, we will develop more fully the concept of natural modes, derive a measure sensitive to whether a set of categories corresponds to natural clusters, and generate a procedure by which the observer can recover the natural categories from the data provided by the environment.

1.3 Thesis Outline

The thesis is logically divided into three parts. The first part develops the philosophical groundwork for the recovery of natural categories. Chapter 2 begins with a discussion of the goals of categorization and how those goals require an appropriately structured world. The Principle of Natural Modes is then developed as a characterization of the structure of the world and as a basis for categorization. The philosophical, physical, and psychological implications of the claim of natural categories are explored; in particular we reconcile formal logical arguments against natural categories with the physical and psychological evidence supporting their existence. Chapter 3 examines some of the previous work in the fields of cognitive science, cluster analysis, and machine learning that is relevant to recovery of natural categories.

The second part of the thesis, consisting of chapter 4, addresses the prob-

lem of measuring how well a set of categories reflects the structure of the natural modes. We develop a measure, based on information theory, that assess how well a set of categories supports the goals of the observer: the reliable inference of unobserved properties from observed properties. Because it is the existence of natural modes that permits the observer to accomplish this inference task, we argue that a set of categories — a *categorization* — that supports the goals of the observer must reflect the natural modes. The behavior of the measure is demonstrated in the natural domain of leaves.

Finally, in chapters 5 and 6, we address the issue of the recovering the natural modes from a set of data. In chapter 5, we define a *categorization paradigm* inspired by the formal learning theory work of Osherson, Stob, and Weinstein [1986]. Within the context of this paradigm, we develop a dynamic categorization algorithm which makes use of the measure developed in chapter 4 to evaluate hypothesized categorizations. The performance of this algorithm is tested in three natural domains, including a set of data that have served as a test for other categorization systems. The results indicate that the categorization algorithm is an effective method for recovering natural categories. An analysis of the competence of the algorithm is provided and predicts the observed behavior.

In chapter 6, we extend the analysis of the categorization algorithm into domains in which there are multiple natural clusterings. Such domains are formed when more than one level process constrains the properties of objects. For example, we will consider the domain of infectious bacteria where there is structure at both the genus and species level. We develop a procedure by which the observer can recover both levels of categories. Furthermore, we provide a method by which the observer can determine which properties of objects are constrained by each level of process. This same mechanism enables the observer to evaluate the utility of a property for performing the categorization task.

In the conclusion of the thesis, chapter 7, we summarize the results of the previous sections, once again consider the utility of recovering the natural categories in the world, and discuss potential extensions to the work.

Chapter 2

Natural Categories

We begin our study of natural object categories by examining a task that explcitly makes use of such categories: object recognition. By recognition we simply mean the act of announcing some category when an object is presented. Our first consideration will be the goal of recognition, which we will propose to be the inference of important unobserved properties from observed properties. If recognition is to be performed by announcing the category to which an object belongs, what kinds of categories would permit the observer to attain this goal? Under what conditions is such a goal possible? To help achieve these goals, we will propose the Principal of Natural Modes: a claim — about the *world* — that there exist sets of natural categories ideally suited to the task of making useful inferences. This claim will need to be reconciled with philosophical and logical arguments against the ontological existence of such categories. In support of natural modes and their use for recognition we will present evidence from both the physical world and the psychologies of various organisms. Finally, we will be able to pose the categorization problem as the discovery of natural mode categories in the world.

2.1 The Goal of Recognition

Suppose we wish to construct a machine (or organism) which is to perform object recognition by announcing some category for each object encountered. What set of categories would be appropriate? Certainly we cannot answer

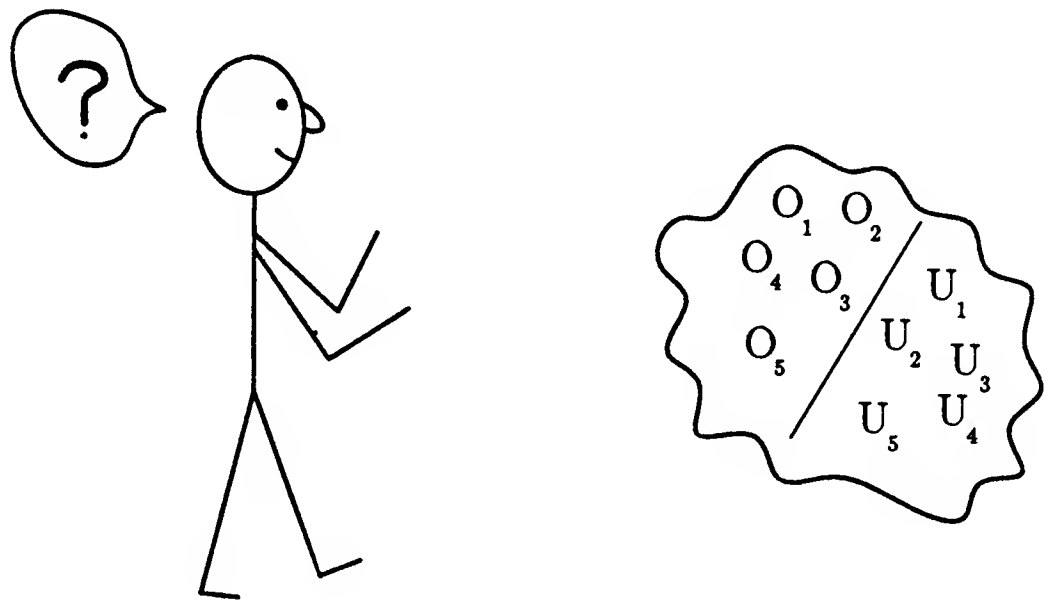


Figure 2.1: A canonical observer viewing a canonical object. The O_i's and U_j's represent observed and unobserved properties, respectively. The goal of the observer is to infer the U_j's from the O_i's.

this question without placing further constraint on the output of this machine. Otherwise, any arbitrary categorization would be valid, e.g. "announce category 1 if the the object is less than 100 feet away; announce category 2 otherwise." Therefore we need to provide an additional constraint as to what makes a suitable or useful categorization.

To provide such a constraint, let us propose that the object recognition task — and therefore object categorization — has as its goal the following:

> **Goal of Recognition** is to predict important unobserved properties from observed properties.

This goal requires that when an object is "recognized," which we have defined to mean when some category is announced, it should be the case that inferences about that the unobserved properties of the object can be reliably asserted. Properties of particular interest are those that affect the object's interaction with the environment, of which the observer is a part.

To illustrate the goal, consider our observer in Figure 2.1. While viewing some object the observer measures certain observable properties O_i. The observed properties may include very simple quantities such as dominant color or overall length, or they may be more complex measures such as a description of the basic geometric parts of the object [Hoffman and Richards, 1985]. From these properties, the observer wants to infer the unobserved properties U_j. These unobserved properties may include function (*"something to sit upon"*) or behavior and affordances [Gibson, 1979] (*"something which moves and will try to eat me"*). This basic inference is really the basic problem of perception, and we can use this goal of recognition to provide criteria for an appropriate set of categories.

Notice, however, that being able to make reliable inferences about an object's properties from its category is not sufficient to satisfy the goal of recognition. Recognition requires using one set of properties (observed) to make inferences about another set of properties (unobserved). Thus, we need not only the ability to infer reliably an object's (unobserved) properties from its category, but also the ability to infer an object's category from its (observed) properties. For example, the validity of the predictions should degrade gracefully as less observed information is provided; it will often be the case that the observer only recovers a subset of the observable properties. Also, the observer should be able to make predictions about objects not previously viewed. That is, the observer must be able to generalize appropriately such that the predictions about the non-observed properties of a novel object tend to remain valid.

As an aside, we should address the (skeptic's) question of why use categories at all to satisfy the goal of recognition. If one's goal is only to make predictions about unobserved properties from the information provided by observed properties, then a more direct strategy would be to recover the relationships between the two. For example, one could estimate all the conditional probabilities (of every order) and use these estimates to make predictions. One response to this argument is that we have not (yet) claimed that categories are the best mechanism for solving the inference problem. Rather, if *given* the problem of constructing categories for the recognition task, then reliable inference is one means of defining suitable criteria. However, we actually do wish to make the claim that categories are an efficient and effective means of achieving the goal of reliable inference about unobserved properties. We must postpone the defense of this claim until we discuss the

principle of natural modes, to be presented in the next section.

Given the goal of constructing a set of categories consistent with the proposed goal of recognition, is it possible for an observer to perform such a categorization of objects? Will his categorization permit the inference of unobserved properties? The answers to these questions clearly depend on the domain in which the recognition system is to operate. If there is no correlation between the sensory data and the behavior of an object, then no such inference is possible. If every object in a world (including witches, bicycles, and trees) is spherical in shape, blue in color, and matte of surface, then such visual attributes would be useless for inferences of unobserved properties important to the observer. Under such circumstances a visual recognition system which performed useful classification could not be built. Therefore, if we are to claim that the goal of the recognition system is to place objects in the world into categories that permit the prediction of unobserved properties, then for such a system to be successful it must be the case that the *world* is structured in such a way as to make these inferences possible. This is a strong claim, and one which is fundamentally different from stating that the only structure present is that which is imposed upon the world by the observer's interpretation.

2.2 Natural Modes

If we take the human vision system as an existence proof that it is possible to define a categorization of objects that permit inferences about an objects unobserved properties (e.g. I can visually categorize some object as a "horse" and predict many of its unobserved properties based upon that categorization), then it must be the case that the natural world is structured in some particular way. What would be the basis of such structure?

To gain insight into this question, consider the Gedanken experiment of giving a grade school art class the assignment of drawing pictures of imaginary animals — animals the children have never seen and about which nothing has been said. The results are as varied as the children who produce them: multiple-headed "monsters", flying elephants, and other composite animals are produced. Completely bizarre-looking creatures also emerge. There seems to be no limit to the the number of animals that one could imagine. Yet, they live only in the mind, and in the world of children's toys

which produce creatures such as Bee-Lions.

If these animals could exist, (i.e. we could physically construct them) why don't they? In some instances, the laws of biological physics simply preclude their feasibility. Flying elephants would require a weight, surface area, and muscle relation that cannot be created from the biological hardware used to make an elephant [McMahon, 1975]. Other animals, although feasible, may not exist because such creatures were either never formed by mutation, or, if formed, they were made extinct by forces in the environment. In this latter case and in the case of impossible animals, we can view the situation as an entity (the animal) which did not satisfy the environmental constraints in effect at the time. In fact, given the complexity of the natural world and the extensive pressures brought to bear by Nature on an organism, most arbitrarily-designed animals would perish, because the chance of creating arbitrary organisms which would be well-suited to the environment is almost zero. Unlike the world of the imagination or children's toys, the natural world cannot contain objects of arbitrary configurations.

As such, the existing species are special in an important way. The species represent finely tuned structures, Nature's solutions to the constraint satisfaction problem imposed by the myriad of negative environmental constraints. "Survival of the fittest" may be interpreted as simply the statement that the surviving species satisfy the environmental constraints better than any other species competing for the same resources. Because of the extent of these constraints, each of the solutions must be highly constrained; that is, there is no small set of properties of an organism which is sufficient for its survival. Stream-lined contours, fins, eyes on opposite sides of their body — these attributes combined with a vast set of internal structures permit fish to survive in the aquatic environment.

Also, these solutions tend to be disparate. [Mayr, 1984; Stebbins and Ayala, 1985]. Because species of similar construction will be competing for the same resources, variations in *properties important to the organism's survival* are removed, unless the variations are large enough such that the organism is now in a different niche. The pressure of natural selection moves the evolution of species to a discrete or clustered sampling along those dimensions relevant to a species survival. We refer to this clustering as the "Principle of Natural Modes," and because it is central to our development of a natural categorization we restate it as follows:

> **Principle of Natural Modes:** Environmental pressures force objects to have non-arbitrary configurations and properties that define object categories in the space of properties important to the interaction between objects and the environment.

We do not live in a world of randomly created objects and visual scenes, but in a world of structure and form.

To refine our claim about natural modes, we let us make explicit the claims that are being made, as well as those that are not. First, the existence of natural modes implies that objects do not exhibit uniform distributions of properties. Rather, objects display a great deal of *redundancy*, redundancy created by the complex sets of constraints acting upon objects. For example, we do not see the mythical Griffin (half eagle, half lion). Objects with beaks also (tend to) have feathers and wings and claws. Redundancies such as these make it "easy" to recognize an object as a bird: a few clues are sufficient. Second, we do not intend to restrict the claim to only natural objects; in section 2.4.1 we will discuss constraints acting on man-made objects as well. Finally, we are not claiming there exists a *unique* set of object categories. We allow for the possibility that the clustering of objects along dimensions important to the interaction between objects and the environment may be "scale" dependent: clustering occurs at different levels of the object hierarchy. For example, consider the division between mammals and birds, and then the separation between cows and mice. The clustering which separates mammals from birds occurs at a level of biological processes much "higher" than that which separates cows from mice. We will further develop the concept of levels of categorization in chapters 4 and 5 when we consider matching the goals of the observer to the structure of the world. For now we can assume that "natural mode categories" refer some selection of categories corresponding to a natural clustering at some level.

In the interest of completeness, two important comments need to be made. The first is that we are not stating that there exist objective categories in the world, *independent of any categorization criteria.* Rather, we are stating that there exists a clustering along dimensions which are important to the interaction between the object and its environment. Therefore, if some sensory apparatus is encoding properties related to these important dimensions, then there will be a clustering in the space defined by that sensory mechanism. The reason for making this point here is that there is a

large body of work by both philosophers and logicians arguing that there do not exist *objective* categories in the world. By restricting the claim to consider only those properties important to the interaction between the object and the environment we can finesse the problem of objective categorization. In section 2.3.1 we will provide a brief review of the arguments against the ontological status of natural categories and we will discuss how those ideas relate to the claim of natural modes.

The second point is that the Principle of Natural Modes is similar to Marr's "Fundamental Hypothesis" which argued that if a collection of certain observable properties tended to be grouped, then certain other properties (unobservable) would tend to group similarly [Marr, 1970]. The principal difference is that Marr did not provide a motivation for why one would expect to find certain observable properties grouped in clusters. In fact, the claim of natural modes by itself is not sufficient to provide a clustering of objects in the feature space of *observable* properties. Therefore we extend our claim with the following addition:

> **Accessibility:** The properties that are important to the interaction of an object with its environment are (at least partially) reflected in observable properties of the object.

Fortunately, this claim is easily justified. For example, the basic shape of an object usually constrains how the object interacts with its environment. The legs of an animal permit it mobility. The color of an object is often related to its survival: plants are green and polar bears are white. As such, the important aspects of an object tend to be reflected in properties which are observable. Therefore, the Principle of Natural Modes taken together with claim of Accessibility provide a basis for why one might expect to find a clustered distribution of objects in an observer's feature space.

Finally we can combine the goal of the observer — to construct a set of categories which allow the observer to predict important unobserved properties of objects — with the claim of natural modes. We make the following claim about the appropriate set of categories for recognition:

> **Natural Categorization:** If an observer is to make correct inferences about objects' unobserved properties from the observed properties, then he should categorize objects according to their natural modes.

This claim follows naturally from our goal of recognition and the proposed Principle of Natural Modes. Given that the observer is seeking to infer the properties which describe how an object interacts with its environment, and given that these properties cluster according to natural modes, then the observer should attempt to categorize objects according to their natural modes. Accessibility states that this goal can be accomplished using sensory data.

Before proceeding to the next sections, let us return to the skeptic's question of why one should use categories to accomplish the proposed goal of recognition — the inference of unobserved properties from observed properties. Now that we have presented the Principle of Natural Modes we can argue that the world contains categories of objects which support generalization. For example, suppose one believes that a certain set of objects forms a natural category, and that one of those objects exhibits a certain (in general) unobserved property, e.g. it attacks human beings. Then, one would make the prediction that all objects of this category would exhibit the same property. If one were using standard conditional probabilities, one could not make this assertion without some particular *a priori* probability statement about how to generalize over objects of "similar" observed properties. But such a statement is equivalent to believing in the existence of natural categories. Thus, a more natural (and more efficient) method of using this knowledge is to explicitly represent the categories themselves.

In the next three sections, we will consider arguments against and evidence for the existence of natural modes. The primary argument against natural modes stems from the work of philosophers and logicians considering the abstract implications of natural categories. The favorable evidence, however, is derived from consideration of the physical world, and the organisms that inhabit it.

2.3 The Philosophical Issue of Natural Categories

2.3.1 Questions of ontology

Ontology may be described as the branch of philosophy that concerns what exists [Carey, 1986]. As mentioned in section 2.2 there has been considerable

attention paid to the question of whether categories can really be said to exist in the world, rather than being constructs in our head. In this section we will provide a brief review of the logical argument against the existence of objective natural categories. Then, we will reconcile this argument with the principle of natural modes.

The basic issue at hand is do categories exist in the world independent of some observer? Would "rabbits" be a more natural category than "round-or-blue-things" if there was no organism to perceive them? Prima facie, the principle of natural modes would argue for the existence of such categories. However, we will see that natural categories can only be said to exist if we provide constraint external to objects themselves; an outside oracle will be required to restrict what aspects of an object may be considered as relevant to categorization. Only then is it reasonable to consider one categorization of objects as more natural than another.

Perhaps the most complete discussion of the subjective nature of categories is provided in Goodman [1951]. There it is demonstrated that, by the appropriate choice of logical primitives with which to describe objects, any similarity relationship between objects can be constructed. Thus, if a natural set of categories is defined by some measure on a similarity metric, then any categorization may be selected. Though thorough, Goodman's presentation is quite dense and difficult to recount. As such we will provide an alternative form of the argument as given by Watanabe [1985]. This formulation — referred to as the Ugly Duckling Theorem — makes the issues of categorization quite clear.

Let us state the theorem directly and then sketch the proof:

> **Ugly Duckling Theorem:** Insofar as we use a finite set of predicates that are capable of distinguishing any two objects considered, the number of predicates shared by any two such objects is constant, independent of the choice of two objects. [Watanabe, 1985, p. 82]

We will provide a proof of this remarkable result for one special case; through it we will be able to see why an external source of constraint is required if we are to consider one categorization more natural than any other.

To prove the Ugly Duckling Theorem, let us consider a world of objects that are described by only 2 binary predicates, A and B (Figure 2.2). In this case the predicates are unconstrained in the sense that A and B carve

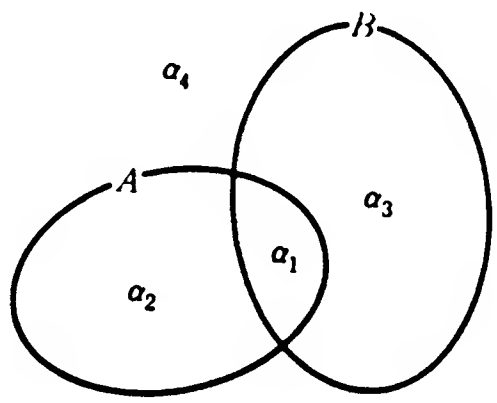


Figure 2.2: A world with two independent starting predicates A and B.

the world up into four different object types, $\alpha_1 \ldots \alpha_4$, corresponding to the the logical descriptions of $\{(A \cap B), (A \cap \neg B), (\neg A \cap B), (\neg A \cap \neg B)\}$. Now let us consider the question of how many properties are shared by any two objects.

First, one must realize that although there are only two starting predicates, there are many composite predicates, and each such predicate is a property in its own right. In fact, every combination of the atomic regions α_i is an allowable predicate or property. Let us define the "rank" of a predicate to be the number of regions or object types (α_i) which must be combined to form that predicate. For example, the predicates of rank 1 are exactly those logical combinations given above. α_1 defines the predicate $(A \cap B)$ which is said to be "true" for the object α_1 and "false" for objects α_2, α_3, and α_4. An example of a predicate of rank 2 is $(\neg A)$ formed by the union $(\alpha_3 \cup \alpha_4)$. An interesting predicate of rank 2 is given by the union $(\alpha_2 \cup \alpha_3)$: the logical equivalent is the exclusive-OR $(A \otimes B)$. The exclusive-OR must be an allowable predicate: if A corresponds to "blind in the left eye" and B corresponds to "blind in the right eye," then $(A \otimes B)$ is the predicate "blind in one eye," a perfectly plausible property. Since all possible combinations

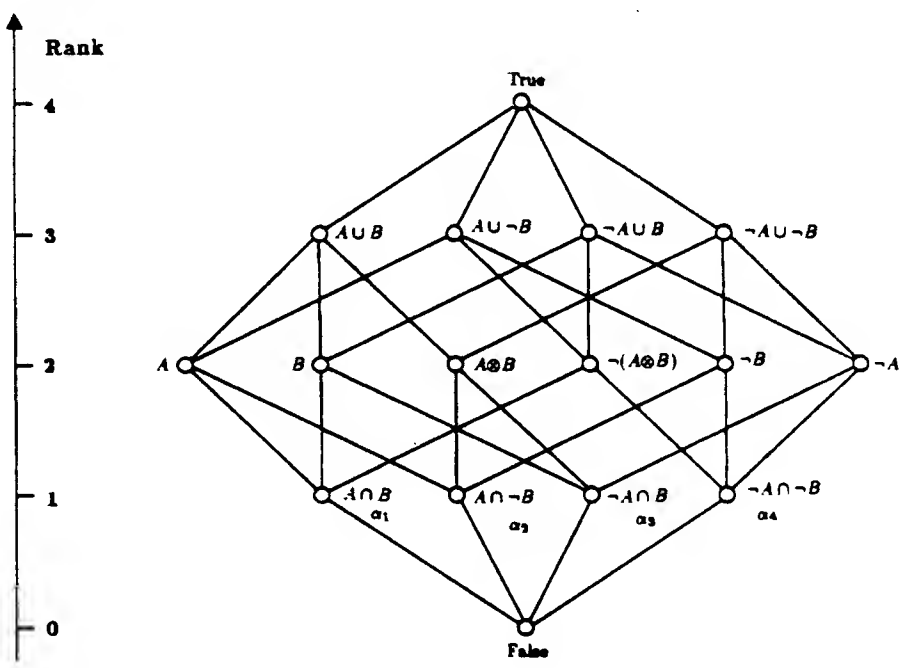


Figure 2.3: Predicates arranged in a lattice layered by rank and connected such that a straight line indicates implication from the lower rank to the higher rank.

of regions are permitted to form predicates (if one allows the null predicate which is false for all objects, and the identity predicate which is true for all objects) there are $2^4 = 16$ possible predicates defined in our simple world of two starting predicates.

We can arrange these predicates in a "truth" lattice as shown in figure Figure 2.3. The lattice is layered by rank and connected such that a straight line indicates implication from the lower rank to the higher one. For example $(A \cap B)$ implies A which in turn implies $(A \cup \neg B)$. Notice that the rank 1 predicates correspond to each of the different possible objects. The properties which are true for an object may be found by following all upward connections from that object's node; similarly, any node in the lattice accessible from two different objects represents a property shared by those objects.

Now, the important question is how many properties are shared by any two objects. Given the symmetry of the lattice is should not be surprising that each of the objects shares exactly 4 properties with each of the other

objects.[1] *If we consider the complete set of possible properties, then any two objects have exactly the same number of properties in common.* Thus any similarity metric based upon on the number of common properties would assign an equal similarity to all object pairs. Given this state of affairs, it would not be plausible to consider any one categorization of objects, any one grouping of instances according to some similar properties, as more natural than some other.

Yet, most observers would agree that a dog and a cat are more "similar" than are a dog and a television. How can we resolve this intuition against the theorem of the Ugly Duckling (so named since it states that the Ugly Duckling is as similar to the swan as is any other duck)? The answer must lie in somehow restricting the set of properties which can be considered. In our simple world of two base predicates there were 14 non-trivial properties which were considered. Under this description all objects were equally similar. If, however, we remove certain properties from consideration, then it will be the case that some pairs of objects will share more properties than others, and a similarity metric base upon shared properties will yield distinct categories. How, then, can we decide which properties to remove from consideration?

Unfortunately, it is impossible to decide which properties to discard simply on syntactic grounds, that is without consideration to their meaning. Both Goodman [1951] and Watanbe [1985] provide persuasive arguments that no property can be regarded as *a priori* more primitive or more basic than any other; a redefinition of terms which preserves logical structure but changes the basic vocabulary can always cause syntactically complicated properties to become simple, and simple ones to become complex.[2] Also, as with the example of "blind in one eye," unusual or disjunctive concepts may be just as sensible as those defined more simply in a given vocabulary. Thus, if we are to weight some properties more than others, we must have an external motivation for doing so. This source of information is referred to as

---

[1] Watanabe [1985] extends the discussion to include any number of predicates. In general if there are m atoms, where an atom is defined by an indivisible region α_i, then there are 2^m predicates and any two objects share $2^{(m-2)}$ of them. This result is valid even if the starting predicates are constrained, e.g. the predicate B includes A such that A implies B. The only critical assumption is that the vocabulary used to describe the objects partition the world into a finite number of distinct classes.

[2] Though see Osherson [1978] on some syntactic conditions which should be met by "natural" properties. This claim, however, is controversial. (see [Keil, 1981; Carey, 1986])

"extra-logical" by Goodman.

Let us once again consider the principle of natural modes. We state that objects will tend to cluster along dimensions important to the interaction between objects (organisms) and the environment. That is, we claim that if we restrict the properties of consideration to those only involved with an object's interaction with the environment, then there will be a clustering of objects which will define natural categories. Thus our external source of information, our oracle which decides what properties should be considered, are the laws of the physical and biological world. The physical constraints imposed upon objects and organisms select the properties of objects in which natural categories are defined.

2.3.2 Induction and natural kinds

A related problem of philosophy is the issue of *natural kinds*. As an illustration, consider an example similar to that described by Quine [1969]: An explorer arrives on an uncharted island, and meets natives never before visited by "civilized" men. Being an amateur linguist the explorer attempts to compile a dictionary of the vocabulary of the natives. One day, while accompanying the explorer on a trip through the forest, a native points to an area where a rabbit is sleeping beneath a tree and utters the word "blugle." The explorer writes in his dictionary that "blugle" means "rabbit." Quine asks how does the explorer know that the native is referring to the rabbit and not the situation rabbit-under-a-tree. Even if the explorer could test this distinction (say by pointing to another rabbit, perhaps cooked, and announcing "blugle" and awaiting the response) he could never test all possible meanings consistent with the situation.

Yet, we believe the explorer is probably correct in his conclusion, and even if he is not correct on his first attempt, we believe that he will probably be correct on his second or third (perhaps "blugle" means "sleeping" or "cute," but surely it does not mean "small-furry-leg-shaped-piece-within-ten-meters-of-that-particular-tree"). After considering how it is possible for the the explorer is likely to be correct, and related problems such as why people tend to agree on the relative similarities between objects, Quine concludes that people must be endowed with an innate "spacing of qualities" [1969, p. 123]. Such a spacing would provide people with a standard of similarity that permitted convergence of their descriptions of the world. An innate quality

space is an example of extra-logical constraint being provided to the observer for the formation of object categories.

2.4 Natural Object Processes

In this section we provide a brief discussion of the physical basis underlying the natural modes. In Bobick and Richards [1986] the construct of an *object process*, is proposed as a model of the processes responsible for the creation of natural modes. An object process represents the interaction between some generating process (which actually produces objects) and the constraints of the environment. For the discussion here we consider some of the physical evidence for natural object processes responsible for the natural modes and relate those processes to the claim of Accessibility.

2.4.1 Physical basis for natural modes

We have made a claim about the structure of the natural world: objects cluster in dimensions significant to their interaction with the environment. If this is the case, then there must be underlying physical processes which give rise to these clustered distributions, and produce these natural modes of objects. Therefore we should be able to find evidence in the world of such processes.

Fortunately, such evidence is quite abundant. In the world of biological objects, the fact that structures must evolve from previous structures places a strong constraint on the forms present [Dumont and Robertson, 1986; Thompson, 1962]. An interesting observation supporting this claim is provided by Stebbins and Ayala [1985] who noted the non-uniformity in the distribution of the complexity of DNA. As Pentland [1986] has noted, "evolution repeats its solutions whenever possible," reducing the number of occurring natural forms; this conclusion was also reached by Walls [1963] in his discussion about the repeated evolution of color vision. Additional support for principle of natural modes comes from the field of evolutionary biology. Mayr [1984] states:

> [The biological species] concept stresses the fact that species consist of populations and that species have reality and an internal genetic cohesion owing to the historically evolved genetic program that is

shared by all members of the species.

The objective existence of species represents a structuring of the world *independent of any particular observer.*

Structure in the physical world can also be discovered by examining the physical processes responsible for the existence of many forms. Steven's analysis of patterns [1974] is an example of constraint imposed by the physics of matter in the formation of structure; the fact that "interesting" patterns emerge is an example of natural modes. (See also Thompson [1962].) The work by vision researchers to model different physical processes so as to construct representations for different types of objects is plausible only because there are limited ways for nature to create objects [Pentland, 1986; Kass and Witkin, 1985; Pentland, 1984]. Even chaotic systems have modes of behavior [Levi, 1986].

It should be noted that man-made objects are also subject to constraints upon form, although the environmental pressures are different. For example, a chair must have certain geometric properties to be able to function appropriately. It must allow access and stability, placing significant constraints on its shape. A table must have a flat nearly horizontal surface with a stable support to function as a table. An even more complicated set of constraints related to ease of manufacturing and peoples' aesthetic interests operates on most constructed objects. Why is it that most books have similar aspect ratios? The common visual scene of "row houses" is an example of structure imposed by man mimicking the type of natural modes produced by nature. For a more extensive discussion about constraints on the shapes of objects and the non-arbitrary nature of objects see [Winston, et al., 1983; Lozano-Perez, 1985; Thompson, 1961].

2.4.2 Observed vs. unobserved properties

It is important to relate the existence of natural object processes to the claim of Accessibility. The claim of Accessibility states that some of the properties important to an object's interaction with the environment are reflected in observed properties; the importance of this claim is that it permits us to attempt to recover the natural categories from the observed properties. In light of the discussion about natural object processes, we can view Accessibility as independence between the sensory processes and the processes responsible for the structure of an object. Because the distinction between

observed and unobserved properties occurs only because of the sensory apparatus, we can look for natural modes in only the observed properties and assume that the modal behavior of the unobserved properties will follow. Because most of the data provided to the observer are observed properties, this dissociation between observed and unobserved properties is essential for recovering natural categories.

2.5 Psychological Evidence for Natural Categories

Until now, our arguments for the existence of natural modes have rested on evidence from the world itself. In particular we have claimed that the physics of our world, including the evolutionary pressures of the environment, cause objects to have non-arbitrary configurations. However, if it is the case that it makes sense to describe our world as having natural categories, and, as we have claimed, that describing the world in terms of these categories permits one to make useful inferences about objects, then we might expect these categories to be manifest in the psychology of organisms that make such inferences. That is, we should be able to detect the presence of natural categories in the mental organization of the world used by different perceiving organisms. Notice that the existence of mental categories does not imply the existence of categories in the world, only that the world is structured in such a way as to permit the formation of visual categories which are useful to observer. Therefore the ability to create such a categorization is a necessary condition for the expression of natural modes in observable properties.

In fact, a wealth of literature exists attesting to the psychological reality of natural categories. Evidence may be found in both cognitive science and animal psychology. In particular the interaction between natural categories and perceptual recognition tasks has been extensively investigated. We present a brief review of the relevant literature, especially as relates to object perception.

2.5.1 Basic level categories

In 1976, Eleanor Rosch and her colleagues published what has become a classic paper in the field of cognitive psychology [Rosch, Mervis, Gray, Johnson,

and Boyes-Braem, 1976].

The principal finding of that work was that people tend to categorize objects in the world at a one particular abstract taxonomic level. This level is operationally defined as the level at which categories have many well defined attributes but at which there is little overlap of the attributes with those of other categories. As an example consider the simple taxonomic relation of "fruit $\rightarrow$ apple $\rightarrow$ McIntosh-apple" where $x \rightarrow y$ means x includes y. In this case, Rosch et al. demonstrated that the preferred level of description is "apple." The reason for this was given to be that few attributes can be assigned to "fruit" relative to the number of attributes assignable to "apple," while the lower level category "McIntosh-apple" is a category whose atrributes overlap extensively with other lower level categories such as "Delicious-apple." The *basic level,* in this case "apple", is that taxonomic level at which category members have a well defined structure (in Rosch's concrete noun examples we explicitly mean physical structure) and at which there were no other categories that significantly share that structure. Perhaps the most important aspect of the work by Rosch, et al. was the demonstration that categories at the basic level appear to be more accessible for a variety of cognitive tasks (presently we will consider the interaction between basic level categories and the perceptual task of object recognition), indicating that these categories enjoy some special psychological status. That is, is there strong evidence that these categories have some degree of psychological reality.

Several attempts have been made to formally define basic level categories in terms of attributes and categories; this thesis implicitly contains one such attempt. Let us postpone the discussion of these theories until chapter 3 where a review of the various disciplines which have addressed the categorization problem — these include cognitive psychology, pattern recognition and machine learning — is presented. For now, the important point is that there exists empirical evidence of a particular set of categories being used to describe objects in the world.

One of the cognitive operations in which basic level categories show a marked superiority is that of object recognition, whether the actual task be a speed of naming task [Rosch, et al. 1976; Murphy and Smith, 1982; Jolicoeur, Gluck, and Kosslyn, 1984;] or a confirmation task where the subject is primed with the name of a category and has to decide whether a picture of an object belongs to that category (see the analysis of Potter and Faulconer [1975] given in Jolicoeur, et al. [1984]). These findings are of particular interest here

because the principal problem addressed by this thesis is that of categorizing objects into classes suitable for the recognition. Specifically, we would like to know whether basic level categories are special in a *perceptual* sense as opposed to simply being more easily accessed as concepts by some cognitive process.

To address this question, Murphy and Smith [1982] designed an artificial world in which to test the perceptual superiority of basic level categories. By using artificially created superordinate, basic, and subordinate categories, they were able to control factors such as word frequency, order of learning, and length of linguistic description (real basic level categories tend to have simple one word labels). These factors were considered to be possible confounding factors in the results originally reported by Rosch, et al. [1976]. Murphy and Smith did indeed replicate the finding that objects can be categorized fastest at the basic level. They attributed this superiority to the fact that basic level categories have more perceptual structure than superordinate categories, while at the same time having many discriminating attributes from other basic level categories. Because these were artificial objects, Murphy and Smith were able to claim that the advantage demonstrated by the basic level categories in the task of recognition was caused by a purely perceptual mechanism.

Jolicoeur, et al. [1984] extended the work of Murphy and Smith. Murphy and Smith [1982] postulated that categorizing objects as belonging to superordinate categories was difficult (slower) because of the disjointedness of the perceptual structure. For example, to test if an object is a fruit would require matching the incoming stimulus to a highly disjunctive perceptual model (something that would match either a banana or an apple). Jolicoeur, et al. make the stronger claim that that superordinate and subordinate categorizations are slower because *object recognition first takes place at the basic level*, and then further processing is required to determine the superordinate or subordinate category. For example, if the task requires determining whether an object is a fruit, then when presented with an image of an apple, the subject would first recognize the object as an "apple," and then use semantic information to conclude that it is indeed a "fruit." Similarly, if attempting to categorize at the subordinate level, the subject would again first determine the basic category and then compute the necessary additional perceptual information required to determine the subordinate level, e.g. "McIntosh."

To test this hypothesis, Jolicoeur, et al. considered the correlation between latencies in both perceptual and non-perceptual tasks. In one experiment they discovered that the time to name the superordinate category of an object when presented with its image correlated well with the time to name the superordinate category given the word describing an objects basic category. For example, the latencies measured when subjects were given the word "apple" and required to announce "fruit" behaved similarly to those latencies recorded when subjects were presented with a picture of an apple. One possible interpretation of this result is that that some words are inherently easier to access than others. To rule out this possibility, correlations were checked for items within the same superordinate category; both "apple" and "banana" require the response "fruit." For each such item the correct superordinate response is identical, allowing us to remove the effect of the degree of difficulty in making the response. Here too the latency of the perceptual task correlated well with the latency of the linguistic task. Thus the superordinate categorization data support the claim that perceptual access does indeed occur at the basic level.

Jolicoeur, et al. [1984] performed a second experiment to test the claim that objects were accessed at the basic level. Recall that under this hypothesis additional *perceptual* processing beyond basic level is required only for subordinate categorization. Superordinate identification required only semantic information (e.g. knowledge that an apple is a fruit). Thus one would expect a differential effect between the latencies (and error rates) of identification for subordinate and superordinate categories as one varied the the duration of exposure to the perceptual stimulus. In fact, such a differential effect was found: reducing exposure times from 1 sec. to 75 msec. produced no effect on the latencies to name superordinate categories but produced a large increase in the time required to name the subordinate category. Thus, the subordinate categorization data also support the claim that object recognition first occurs at the basic level.

In summary, cognitive psychology provides evidence that people make use of a particular categorization of the world in a variety of cognitive tasks. These basic level categories occurred at the taxonomic level at which objects possessed a high degree of structure while minimizing category overlap; this condition is equivalent to stating that knowledge of an object's basic level category would permit many inferences about the objects properties, while identifying an object's category would be reliable given the minimal overlap

with other basic level categories. While the existence of these categories does not necessarily (in the logical sense) imply the existence of natural categories in the world, it does support the view that the world is structured in such a way as to make a categorical description useful a variety of tasks. The work demonstrating that object recognition first takes place at the basic level supports our claim in section 2.2 that categories which would be useful for making reliable inferences about objects are the appropriate categories for recognition.

2.5.2 Animal psychology

If the structure of the world is such that there exists a categorization which is natural for recognition (would permit reliable inferences about objects) then it should be the case that other organisms in the same world would also exhibit such a categorization in their psychologies. Therefore let us consider the work performed with animals in trying to establish which set of categories they possess. Unfortunately one is limited in the types of tasks one can require an animal to do, and most conclusions about animals' categories are based on how well and how quickly they learn to discriminate various sets of stimuli. Nevertheless, interesting results about the categorization of objects used by animals have been reported. Hernstein [1982] provides an excellent review of the studies of animals' categories.

Cerella [1979] studied the ability of pigeons to learn to discriminate white-oak leaves from other types of leaves. After learning to perfectly discriminate 40 white-oak leaves from other leaves, the pigeons were able to generalize to 40 new instances of white-oak leaves. Such results suggest that the pigeons acquired a "category" corresponding to white-oak leaves. Cerella then trained pigeons using 40 non-oak leaves and *one* white-oak leaf, repeated many times; he then tested these pigeons with probes including 40 different white-oak leaves. Still, with only having seen one white-oak leaf, the pigeons were able to successfully discriminate between white-oak leaves and other leaves. This remarkable finding suggests that not only do the pigeons form a category corresponding to the white-oak leaves, they also extract the attributes necessary to distinguish the "natural" category white-oak leaf from other leaves. This type of learning provides powerful evidence that the world is clustered in recoverable natural modes: an organism's perceptual processes are tuned to be sensitive to the attributes of objects that are

constrained by the processes responsible for the object's formation. As in the experiments reported by Gould and Marler [1987] concerning the role of instinct in animal learning, these results underscore the importance of providing the organism with the necessary underlying theory of structure if the organism is to successfully interact with its environment.

Hernstein and de Villiers [1980] tested the ability of pigeons to learn the "natural" category of fish. One of the reasons they chose fish is that fish are *not* part of the natural habitat of pigeons and thus their prior experience could not influence the results. Their training stimulus set consisted of 80 under-water photographs, 40 which contained fish (in various orientations and occlusion relations) and 40 which did not; the negative examples did contain images of other creatures such as turtles and human divers. Pigeons rapidly learned to discriminate between the two sets of images, reaching a rate of discrimination comparable to that of experiments using objects normally found in the pigeons habitat such as trees and people [Hernstein, Loveland, and Cable, 1976]. When tested on novel pictures, all the pigeons generalized in at least some of the tests. Another set of pigeons was trained using the same stimuli, but, in this case the pictures were divided randomly. The pigeons were unable to achieve a discrimination ability comparable to the fish versus non-fish group and any ability they did acquire took longer to achieve. Thus we may take these findings to suggest that pigeons developed the "natural" category of "fish." The interesting aspect of this result is that fish are not part of the environment normally experienced by pigeons nor have they been so for 50 million years. Therefore it is unlikely that the genetic experience of the species would encode the category "fish." Thus we can assume that there is something about the general perceptual process of the pigeons which makes "fish" a natural category. This is analogous to Quine's [1969] innate quality space mentioned in section 2.3. The fact that the innate quality space of pigeons — an organism unfamiliar with the aquatic environment — would lead to the formation of a category "fish" is additional evidence that natural modes exist in the world and that they are perceptually recoverable.

Chapter 3

Previous Work

Although the problem of categorization addressed in this thesis is one of psychology — how do people organize their representation of objects with respect to recognition — the general problem of discovering "natural" or important classes in a collection of instances can be found in many branches of science. Particularly relevant here are the following three disciplines: 1) cognitive science, in which several attempts have been made to formalize the concept of basic level categories; 2) cluster analysis, the study of the automated partitioning of numerical data into meaningful classes; and 3) machine learning, a subfield of artificial intelligence which considers the issues involved in producing a machine which can learn about structure in its environment. The scope of this chapter precludes giving a thorough description of all the relevant work contained in these disciplines; several complete books have been dedicated to each. As such, we will present a brief description of the important contributions which relate directly to the problem addressed by this thesis: discovering a set of categories that are useful for recognition in terms of permitting reliable inferences about an object's properties. The reader is referred to [Smith and Medin, 1981], [Anderberg, 1973], and [Michalski, et al., 1983; Michalski, et al., 1986] for references giving more detailed analyses.

3.1 Cognitive Science

In section 2.5.1 we referred to the work on basic level categories as evidence for the existence of natural categories in human mental representation; we now consider that work in terms of its theoretical development. Cognitive scientists have attempted to formalize the definition of basic level categories in terms features and their distributions. Because these categories display the desirable properties discussed in chapter 2 — they are highly structured permitting many inferences to be made about the properties of objects contained in those categories, and they are quite dissimilar to one another making classification more reliable — it is important to understand these prior attempts to specify basic categories. We will then draw upon some of them in our own development of a categorization metric in chapter 4.

In the original work of Rosch, et al. [1976] basic level categories are described as the taxonomic level which maximized the *cue validity* of a category. As used by Rosch, et al., the cue validity of a feature for a category is a psychological quantity which measures how indicative a certain feature would be of some category. The cue validity of a category is defined to be the sum of the cue validities of the various features or attributes true of the objects in that category. For example, the cue validity of feathers would be very high for the category "birds," but less so for "ducks" since many objects which have feathers are not ducks. Likewise for the features "wings", "beaks", and "lays eggs". To consider whether basic level categories can be defined in terms of cue validity we need to provide a formal description of that psychological quantity.

The most common formal definition of cue validity is that of *conditional probability.*[1] That is, the cue validity of some feature f_i for a category C_j is

[1] Unfortunately, the term "cue validity" has more than one formal definition in the cognitive science literature. Conditional probability is the interpretation taken by Smith and Medin [1981] and Murphy [1982], though the formulation provided by Smith and Medin (p. 79) is mathematically incorrect. The cue validity to which Rosch, et al. refer is probably based upon the definition provided by Beach [1964] and Reed [1972]. In their formulation, the cue validity of a feature for a category is calculated by considering both the frequency of occurrence a feature (averaged over all categories) and its diagnostic value in identifying that category. Let p be inversely proportional to the over-all frequency of occurrence of some feature f_i. Then, in this formulation, the cue validity of feature f_i for some category C_j is equal to $p \cdot (\text{prior probability } C_j) + (1-p) \cdot (\text{conditional probability } C_j | f_i)$. This formulation was provided to explain the psychological phenomenum that subjects

taken to be the conditional probability that some object is a member of C_j if it is known that feature f_i is true of that object. (We assume for now that a feature is either true or false for any given object.) A simple formula for the conditional probability can be given in terms of the frequency of occurrences of a feature in different categories. Let us assume that N_a is the number of objects in the category C_a for which some feature f_i is true. Likewise for N_b, and for now we can assume that there are only two categories. Then, if we assume that the number of occurrences can be used to estimate the underlying probability distributions, then simple probability theory yields:

$$\begin{array}{l}\text{Cue validity of} \\ f_i \text{ for category } C_a\end{array} = P(C_a|f_i) = \frac{N_a}{N_a + N_b}$$

If there are more than two categories, then the additional occurrences of the feature in those categories are simply added to the denominator. The denominator is simply the total number of objects in the world exhibiting the feature; it remains constant regardless of the number or type of categories into which the world is partitioned. With this definition in hand, we can now consider whether basic level categories can indeed be defined in terms of cue validity.

As Murphy [1982] has noted, maximizing cue validity *cannot* be the basis for basic level categories. A simple example will quickly demonstrate this fact. Following Murphy, let us consider the taxonomic hierarchy of "physical-object", "animal", "bird", and"duck", and let us examine the cue validity of the feature "has-wings" with respect to this hierarchy. Again, define N_{phys} to be the number of "physical-objects" for which the feature "has-wings" is true. Similarly for N_{animal}, N_{bird}, and N_{duck}. By definition, $N_{phys} \geq N_{animal} \geq N_{bird} \geq N_{duck}$. Therefore, since the denominator in the expression for cue validity remains constant regardless of the partitioning of the objects, it must also be the case that the cue validity for "has-wings" increases as one moves up the taxonomic hierarchy. This agrees with the intuition that if p is the probability that some object is a "bird" given that one knows some feature about that object, then the probability that it is an "animal" should be at least p. Since the cue validity of any feature for a category increases as the

---

tend to weight the true conditional probability of feature by how often the feature tends occur. However, this formulation must be considered ad hoc, motivated only by a desire to fit the data. See Kahneman and Tversky [1980] for a detailed discussion about the relationships between probability theory and people's predictive judgments.

category becomes more inclusive, the most inclusive category would be the level which maximized total cue validity. The basic level categories would possess lower cue validities than the most general category "thing."

The underlying reason that cue validity cannot be used to define basic level categories is that cue validity contains no consideration of the density of a feature within a category. That is, only the conditional probability of the category given the feature is measured, ignoring the likelihood that a given category member contains that feature. This extra component is included in the *collocation* measure proposed by Jones [1983]. Using the above notation, the collocation of a category and a feature K_{C_j,f_i} is defined by:

$$K_{C_j,f_i} = P(C_j|f_i) \cdot P(f_i|C_j)$$

The first term is the conditional probability corresponding to the cue validity discussed above. The second term, however, reflects the density of a feature in the category. Because the collocation is a product of these two probabilities, it's value can be large only when both terms are large. Though the first term (cue validity) grows as categories become more inclusive, the second term is diminished when a category becomes less homogeneous. Thus the maximum of this function will occur at some intermediate depth in the taxonomic hierarchy. Jones argues that the basic level categories occur at the taxonomic level which tend to maximize the collocation as measured over all the features.

A simple example will illustrate the properties of the collocation measure and how it relates to basic level categories. After Jones [1983], suppose we have the feature *can-fly* and the hierarchy "duck", "bird", "animal". Suppose there are 10 instances of "duck", all of which can fly, 90 other instances of "bird", 80 of which can fly (allowing for some non-flying birds) and 900 additional instances of "animal" of which 10 can fly (for animals such as bats). If we assume that the occurrences can be used to estimate the probabilities, then we can compute the following collocations: $K_{duck,can\text{-}fly} = .10$, $K_{bird,can\text{-}fly} = .81$, $K_{animal,can\text{-}fly} = .10$. Thus, the collocation measure attains a maximum at the basic level ("bird").

Jones [1983] proposed a particular method for converting raw collocation measures into an index measuring the degree to which a category is basic. This construction can only evaluate one category with respect to the other categories of some categorization. It does not readily permit one to compare one set of categories to another, making it inadequate for the task of selecting

an appropriate set of categories. Also, there is a question as to whether the "degree to which a category is basic" is a meaningful quantity. However, the basic principal of combining two terms reflecting the diagnosticity of features and the homogeneity of categories (here expressed as conditional probabilities) is consistent with the goals of categorization proposed in chapter 2. A high degree of homogeneity in a category permits an observer to infer features (properties) of an object once its category is known, and a high diagnosticity of a feature for some category makes correct categorization easier and more reliable. In chapter 4, where we develop a measure of the utility of a set of categories for recognition, we will return to this discussion of these two, in some sense opposing, goals.

It should be noted that in terms of being useful for our purposes of creating a categorization which is suitable for recognition, there is a fundamental difficulty with the collocation measure: the relative weights of the two probabilities are arbitrarily set to be equal. To examine this issue more closely, consider the following modified version of collocation:

$$K'_{C_j,f_i} = P(C_j|f_i)^{\lambda} \cdot P(f_i|C_j)^{(1-\lambda)}$$

In K', the exponent λ, $(0 \leq \lambda \leq 1)$ reflects the relative contribution of the ability to infer an object's category given its features (expressed as the conditional probability $P(C_j|f_i)$) as compared to the ability to predict an object's features given its category ($P(f_i|C_j)$). Such a relative weight is necessary if we are to use this measure to help select a categorization appropriate for recognition. The observer needs to be able to trade-off how much information about an object he needs to infer from the category against how difficult it is to identify an object's category from its features. Without such a parameter, the categories that the collocation measure will select as basic or fundamental will be completely determined by the distribution of features which the observer measures; the goals of the observer cannot be used to constrain the selected categories. In chapter 4 we will introduce an explicit parameter which represents this trade-off.

Finally, we should mention the work of Tversky [1977]. In that seminal paper, Tversky constructs a *contrast* model of similarity; it is so termed because the similarity between two objects depends on not only the features they have in common, but also the (contrast) features they do not. By further introducing an asymmetry in the manner in which two objects are compared, Tversky is able to explain the empirical finding that similarity

is not psychologically symmetric. For example, most subjects rated North Korea as more similar to China than China was to North Korea (Remember these data were recorded in 1976!). The aspect of the contrast model theory which is relative to our discussion is Tversky's proposal of using his similarity metric as a measure sensitive to the basic level categories. Tversky realized that a measure which only considered the similarity between members within a category would select minimal categories (the opposite of cue validity); categories would tend to become more homogeneous as they were refined. To compensate for this deficiency Tversky suggested creating a measure to select basic categories by multiplying the average similarity between objects in a category by a weighting factor which increased with category size. This product would then behave in a fashion somewhat analogous to that of collocation. However, a weight based upon category size must be viewed as an ad hoc solution; the number of objects contained in a category should not determine whether that category is at the basic level in a taxonomy.

We should note that cognitive science has not addressed the question of how basic level categories are acquired. That is, even if one has a measure which is sensitive to the basic level of a taxonomy, one cannot recover the basic level categories unless a taxonomy is provided. Arguing that a taxonomy is provided through instruction (objects are placed in a hierarchy by teachers) seems to be an untenable position; otherwise, one would have to believe that in the absence of instruction basic level categories would not be formed. Also, the fact that animals form "natural" categories about objects with which they (and their ancestors of 50 million years) have had no experience [Hernstein and de Villiers, 1980] argues against taxonomies provided by instruction.

In summary, we can conclude that a measure which is sensitive to basic level categories must contain at least two components. These components should reflect not only the similarity within a category, but also the dissimilarity between categories. (In the next section we will see that these two components are key to many cluster analysis algorithms.) However, providing a measure which can indicate basic or natural categories is only part of the categorization problem. The issue of how one *discovers* these categories, of how one hypothesizes about which categories are natural, must also be addressed.

3.2 Cluster Analysis

One of the common problems encountered in science is that of generating a reasonable interpretation and explanation of a collection of data. Whether the data consist of various astronomical recordings [Wishart, 1969] or of the descriptions of several diseased soy bean plants [Michalski and Stepp, 1983b], a basic step in the analysis of the data is to appropriately group the data into meaningful pieces. In the case of the astronomical data, the spectral-luminosity profiles are grouped in such a way so as to identify four classes of stars: giants, super-giants, main sequence stars, and dwarfs. This grouping process — segmenting the data into classes which share some underlying process — is often the most important and yet the most difficult step in any experimental science.

Cluster analysis (sometimes referred to as unsupervised learning) is the study of automated procedures for discovering important classes within a set of data. Traditionally, data are represented as points in some d dimensional feature space, where each dimension is some ordinal scale. Such a representation allows one to construct various distance metrics, and then to use those metrics to define "good clusters." Algorithms are then developed to discover such clusters in the data. We present of brief analysis of common metrics and methods used in cluster analysis, and will relate these comments to our current question of object categorization for recognition. The presentation here is drawn in part from Duda and Hart [1973] and Hand [1981].

3.2.1 Distance metrics

At the heart of every clustering algorithm lies a distance metric which defines the distance between any two data points. Most of these metrics require that the data be represented as points in an d dimensional space, and that distances along each dimension be well defined.[2] Standard numerical axis are

[2]Some approaches to cluster analysis have defined distance metrics on representations which use binary (as opposed to ordinal) dimensions (see for example Jardine and Sibson [1971]). The distance between two objects is defined to be the Hamming distance: the number of dimensions on which the objects take different values. The similarity between two objects — the logical inverse of distance — is referred to as the *matching coefficient*. These metrics, however, have difficulties similar to those associated with traditional distance metrics (see text). Problems of scale become problems of resolution and relative

typically used in real applications [Hand, 1981]. The notation we will adopt is each object or data point is represented by a vector $\mathbf{x} = \{x_1, x_2, \ldots, x_d\}$ where x_i is the value for $\mathbf{x}$ in the i^{th} dimension.

An important question in designing a distance metric for such a system is whether the measure should be scale invariant. For example, one can assume that the values along each dimension are normally distributed; scaling would then consist of a linear transformation of each dimension to yield unit variances. The difficulty in deciding whether such scaling should be performed is illustrated in figure Figure 3.1. Here, a rescaling of the dimensions changes the apparent clusters. If a measure were scale invariant, it would not be able to detect the differences between these two data distributions. Whether this behavior is desirable depends on the domain and the semantics of each of the dimensions. That is, one cannot decide on the basis of the data alone whether scaling is appropriate. This requirement for outside information is similar to Watanabe's argument against natural categories presented in chapter 2: knowledge of which features are important cannot be determined by looking only at the data itself without additional information being provided.

Ignoring the issue of scaling, we can consider several distance metrics which assume that the dimensions are appropriately scaled.[3] One common distance metric is the usual Euclidean metric:

$$d_1(\mathbf{x}, \mathbf{y}) = \left[\sum_{i=1}^{d} (x_i - y_i)^2\right]^{1/2}$$

By using the Euclidean measure, one is making the assumption that different dimensions are compatible and that distances along a diagonal are equivalent to distances along a dimension. Often, such an assumption is unreasonable: combining years-of-education and height yields no meaningful quantity. In such cases, the *city block* metric is more appropriate:

$$d_2(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{d} |x_i - y_i|$$

In this case the dimensions are weighted equally, but no interpretation is given to the interaction between dimensions.

---

importance; other issues concerning the use of such metrics remain the same.

[3]As such we will not describe such classic measures as Mahalanobis's distance, which assumes the data are sampled from a multi-variate normal distribution and scales the data by the inverse of the estimated cross correlation matrix.

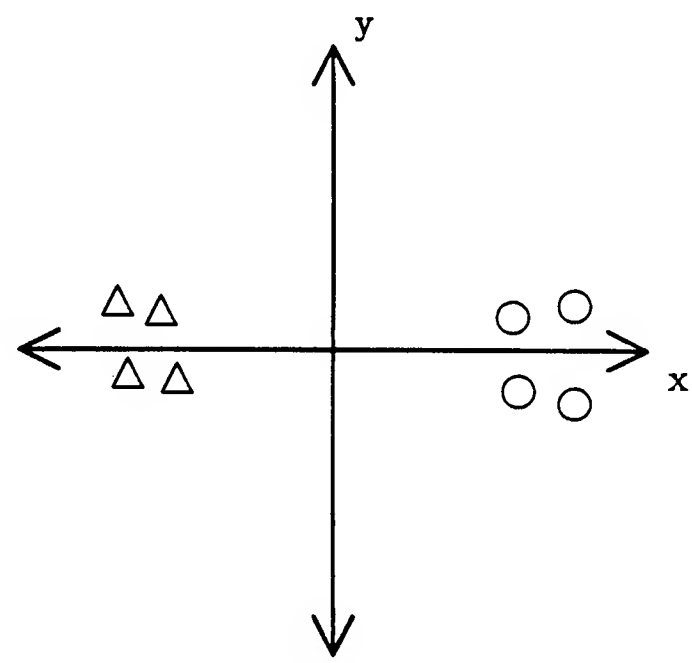


Figure 3.1: By scaling both axis, the apparent clusters can change.

Both $d_1$and d_2 are special cases of the general *Minkowski* distance:

$$d_3(\mathbf{x}, \mathbf{y}) = \left[\sum_{i=1}^{d} |x_i - y_i|^r\right]^{1/r}$$

By varying r one can control the degree of interaction between dimensions. When $r = 1$, The Minkowski measure equals the city block metric; $r = 2$ yields the Euclidean measure. As $r \to \infty$, the Minkowski distance converges to

$$d_4(\mathbf{x}, \mathbf{y}) = \max|x_i - y_i|$$

which represents "total" interaction: the dimension along which two data points differ the greatest completely dominates the other dimensions.

There are two fundamental assumptions in distance metrics such as these whose validity is questionable if the task is one of categorizing objects into categories suitable for recognition. The first of these is the assumption that there are no (or at most few) dimensions that are unconstrained in the data. If there are many such dimensions, then the distances between objects in these dimensions will act as noise, making it difficult to detect the important distances along the constrained dimensions. When attempting to categorize objects for recognition, the important properties — properties which are indicative of an object's category — are as yet unknown. Thus, it is likely that some of the properties measured will be unconstrained in the objects.

The second basic assumption is that the same distance metric is applicable throughout all of feature space. Normally, these distance metrics are insensitive to the absolute values of the feature vectors being compared; the distances between data points are determined solely by the *differences* along each dimension. Thus, these metrics do not alter their behavior as a function of a feature vector's position in feature space. With respect to categorization, this assumption requires that the properties that are important for measuring the distance between some particular pair of objects must be important for all pairs of objects. This requirement does not seem reasonable for a world in which the constrained properties of objects vary from one object type to another.

Finally, most clustering algorithms require being able to specify not only the distance between two data points, but also the distance between a data point and a cluster of data points; the distance between two clusters is often required as well. Because the measure of distance between clusters is often

constrained by the algorithm used for discovering clusters, we will present the inter-cluster measures in those sections discussing clustering methods.

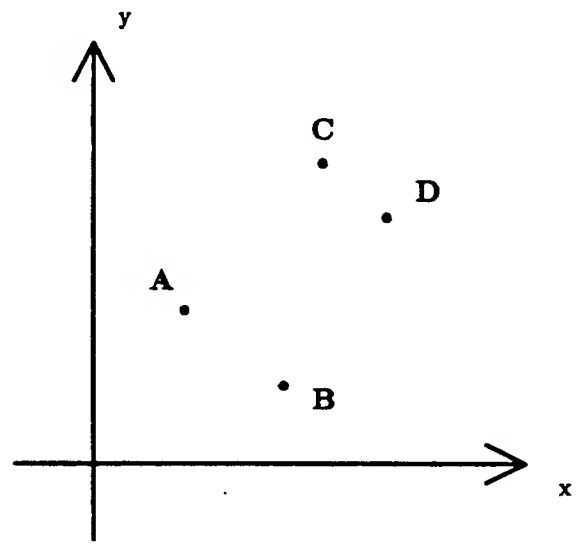
3.2.2 Hierarchical methods

Most cluster analysis programs can be described as being one of two types of algorithms, or as being a hybrid of the two. The first of these consists of *hierarchical* methods which automatically produce a taxonomy of the data samples. In *divisive* clustering, the taxonomy is constructed by starting with all data points belonging to a single cluster and then splitting clusters until each object is its own class. *Agglomerative* methods begin with each sample as a separate cluster of size one and then merge classes until all samples are in one category. Since similar issues underlie both techniques we will consider only the agglomerative methods. Our discussion will follow that of Duda and Hart [1973].

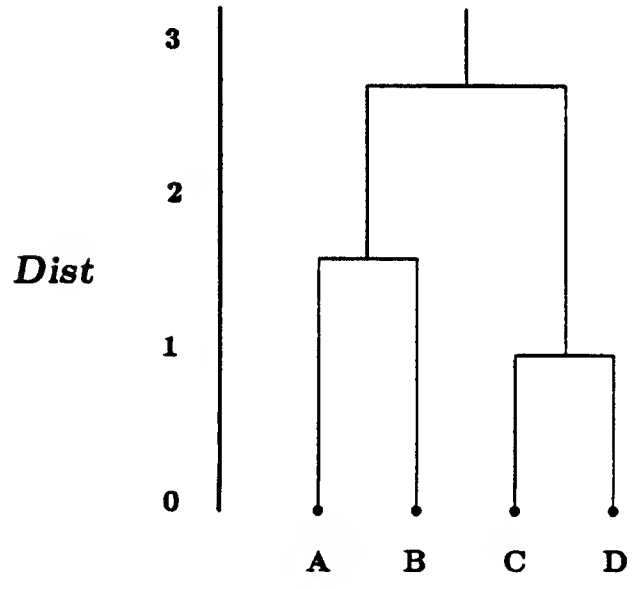
For this discussion, $\hat{c}$ represents the number of clusters; $\mathcal{X}_i$ is the i^{th} cluster, a set of data points; $\mathbf{x}_j$ is the j^{th} data point, represented as a feature vector; n is the number of data points. The basic algorithm for agglomerative clustering can be written as a simple iteration loop:

1. Let $\hat{c} = n$ and $\mathcal{X}_i = \{\mathbf{x}_i\}$, $i = 1, \ldots, n$.
2. If $\hat{c} = 1$, stop.
3. Find the nearest pair of distinct clusters, say $\mathcal{X}_i$ and $\mathcal{X}_j$.
4. Merge $\mathcal{X}_i$ and $\mathcal{X}_j$ into a new $\mathcal{X}_i$, delete $\mathcal{X}_j$, and decrement $\hat{c}$.
5. Go to step 2.

When executed, this procedure produces a *dendrogram* such as that in Figure 3.2. The vertical axis measures the distance between the clusters as they are merged. At a distance of 1, objects C and D were combined to form a new cluster; likewise at a distance of 1.5, A and B were combined. Finally at a distance of 2.5, the cluster {A, B} and the cluster {C, D} were combined to yield a single cluster. In a moment, we will show that the dendrogram does *not* always yield a tree structure which is consistent with increasing distance.



(a)



(b)

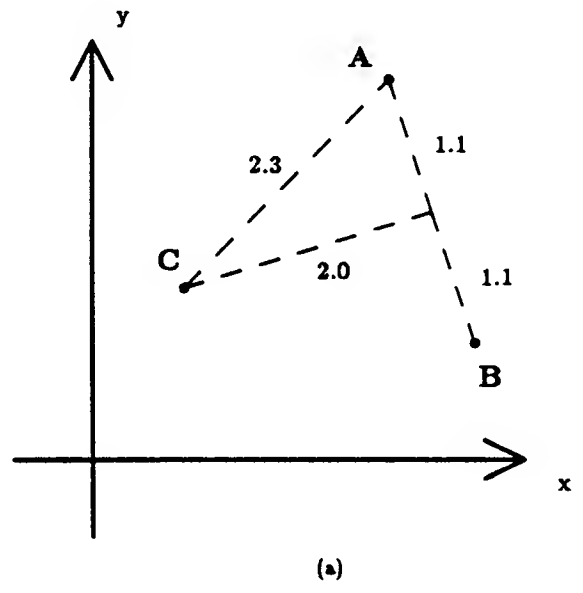
Figure 3.2: Dendrogram (b) of the clustering of the data in (a). This dendrogram would result for many inter-cluster distance metrics including nearest-neighbor and centroid-distance.

To execute the agglomerative procedure, one must define the distance between two clusters. Many measures have been proposed, but we can separate them into those which find a minimal or maximal distance between all possible pairing of objects in the two clusters, and those which compute some average distance. An example of the former group is the *nearest-neighbor* metric in which the distance between two clusters is defined to be the distance between the two closest points, one from each cluster. Because of the ability of a single data point to dramatically affect the distance between two clusters, this class of measures exhibits the undesirable behavior of being sensitive to outlying or "maverick" members in a cluster.

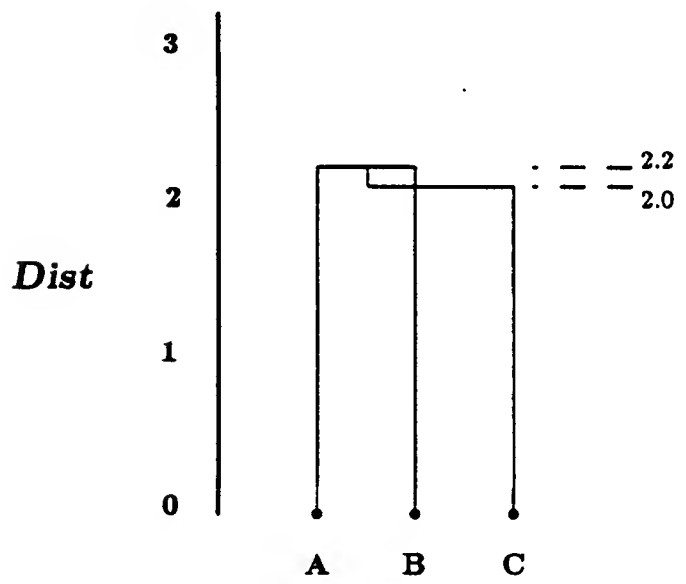
To remove this undesirable behavior, measures based upon either average distances or the distances between average members are used. However, these metrics can cause clusters to be formed that are "closer" to each other than the sub-clusters from which they were originally formed. An example of this is shown in Figure 3.3a. In this case we assume that the distance metric used between two clusters is the Euclidean distance between the (arithmetic) average of each cluster. In this example, data point A is merged with data point B because they are the closest pair; the distance between them is 2.2 units. (Note that A, B, and C could be the average of previous clusters found as opposed to being single data points.) Next, data point C is merged with the new cluster {A, B} as they are the only remaining two clusters. But, the distance between these to clusters is only 2.0 units, less than the original distance between A and B. Thus, the dendrogram displaying this agglomerative clustering might be drawn as in Figure 3.3b; the taxonomy is no longer consistent with distance.

For the task of partitioning objects into categories suitable for recognition, hierarchical methods have a serious deficiency: they require the complete data set be present at the start of the procedure. The addition of a new data point can radically alter the structure of the dendrogram by providing a new link between previously separated clusters. This is especially a problem for methods which rely on an inter-cluster distance metric such as nearest neighbor. Such a system must recompute the entire dendrogram when new data are observed. Because the observer in the natural world will often encounter new objects, a hierarchical approach would not be appropriate for creating natural object categories.

We conclude our description of hierarchical methods by commenting on



(a)



(b)

Figure 3.3: (a) Data points A, B, C with Euclidean distances between them as indicated. The distance between the average of {A, B} and C is less than the distance between A and B. (b) The resulting dendrogram.

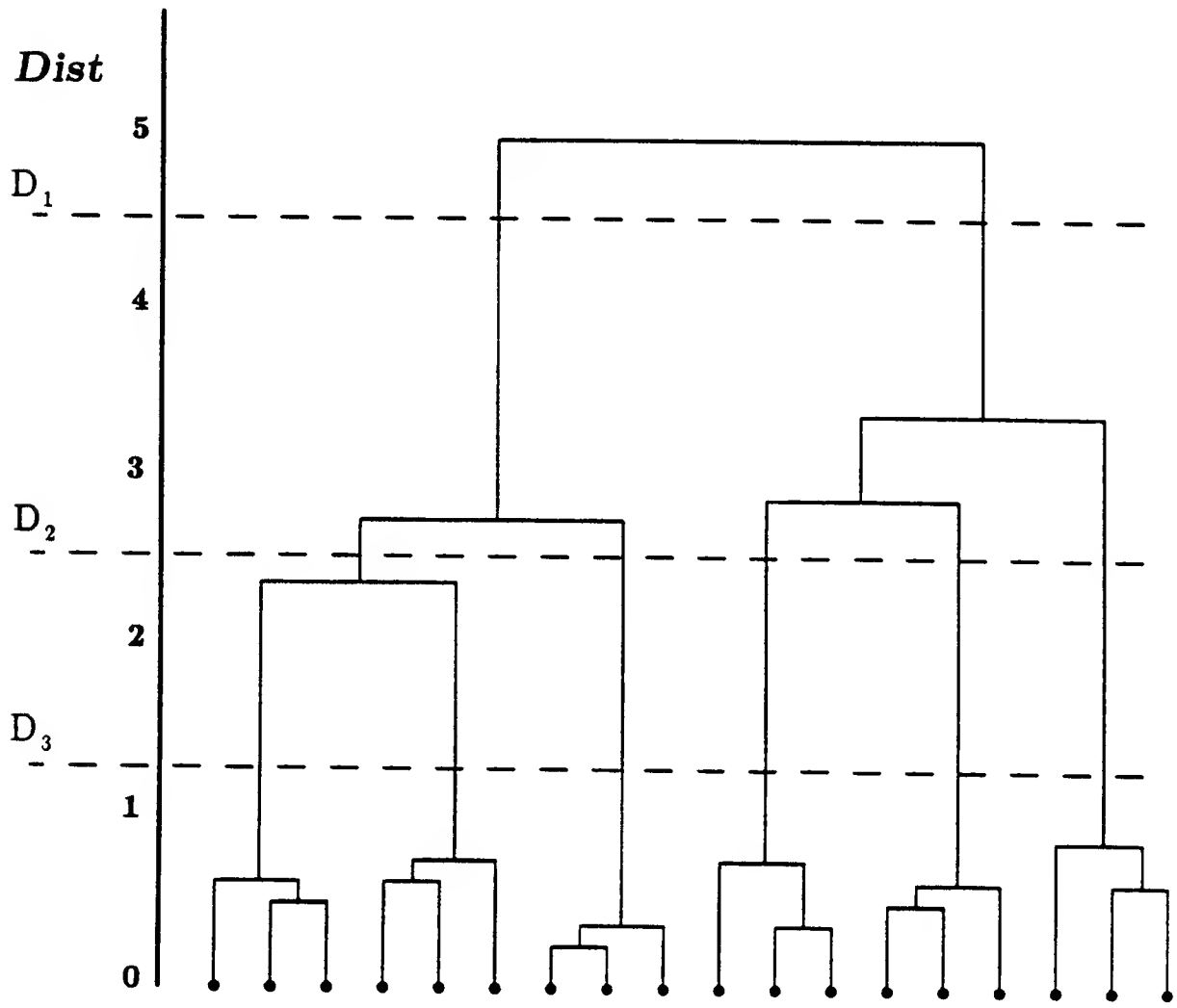


Figure 3.4: A hypothetical dendrogram. If there is some physical significance to the distance measure, one could infer that this data was generated by several discrete processes. In particular, while a description of the data as having 6 groups or 2 groups seems reasonable, a description which claimed there were 5 groups present seems arbitrary. This requirement that the description of the clusters be stable with respect to the distance metric is analogous to Witkin's discussion of the scale-space description of image intensity profiles [Witkin, 1983].

the utility of the dendrogram. Suppose one were to hierarchically cluster some data and that the resulting dendrogram was that of Figure 3.4. Notice that as the distance between clusters is increased the objects are quickly clustered into 6 groups. Then, after increasing the distance sufficiently, 3 of the groups are merged in quick succession while the others clusters remain separate. The process repeats for the other set of 3 clusters. Eventually, the 2 clusters are combined to form one global category. An intuitive interpretation of such a dendrogram is that there are discrete processes reflected in the data and that any valid category description would reflect these processes. For example, description D_2 which partitions the objects into 5 categories seems less valid than either description D_1 or D_3 because of the sensitivity of D_2 to the distance metric. If one thinks of the distance metric as the scale at which the data are observed, then D_1 and D_3 are stable with respect to small changes in that scale, whereas D_2 is not. Zahn [1971] used a similar principle in recovering clusters by dividing a minimal spanning tree graph of the data at edges whose length were inconsistent with the other edges in the tree. The notion of a description being stable with respect to a scale parameter is reminiscent of Witkin's [1983] scale space description of image intensity profiles. In chapters 4 and 6 we will return to this question of stability of description with respect to the "scale" of the observation.

3.2.3 Optimization methods

The second basic approach to cluster analysis is *category optimization.* In these methods, one assumes that there exists some known number of classes c. The data are first partitioned into c classes (either randomly or by some hierarchical method), and then some suitable clustering criterion is optimized by transferring data samples from one cluster to another. An example of such a method is the k-means method which can be written as (following Duda and Hart [1973]):

1. Choose some initial values for means $\hat{\boldsymbol{\mu}}_1, \ldots, \hat{\boldsymbol{\mu}}_c$.

2. Classify the n samples by assigning them to the class of the closest mean. (This is equivalent to clustering the objects to minimize the sum of the squares of the distances of the data points to the cluster means $\hat{\boldsymbol{\mu}}_i$.)

3. Recompute the means $\hat{\mu}_1, \ldots, \hat{\mu}_c$ as the average of the samples in their class.

4. If any mean changed value go to Step 2; else STOP.

Each iteration improves some measure (in this case the sum of the squared distances from the data points to the cluster means) of the "goodness" of the clusters.

As in all optimization procedures, there are two important components to the algorithm. The first component is the criteria used to measure the quality of clusters. Most criteria are based on the *scatter matrices* **W** and **B**, representing the within-cluster scatter and between-cluster scatter, respectively. The formulas for these matrices are unimportant for the discussion at hand; they may be found in Duda and Hart [1973, p. 221]. Their basic purpose is to measure the compactness of each cluster and the inter-cluster separation. Several criteria which attempt to "minimize" **W** (in terms of either the trace or the determinant) and "maximize" **B** have been proposed. The above algorithm which attempts to minimize the squares of the distances between the data points and their cluster mean is equivalent to minimizing the trace of **W**. The use of these matrices reveals the underlying assumption of these measures that "good" clusters are those which are tight hyper-spheres in features space, separated by distances that are large with respect to their diameters. Whether such measures are appropriate for a given task depends upon the validity of the distance metric. Almost all analyses using such scatter matrices assume a Euclidean metric; as discussed in section 3.2.1 such a metric may be inappropriate for object categorization.

It is important to note that categories which can be represented as tight hyper-spheres in feature space begin to satisfy the criteria for a categorization proposed in chapter 2. If categories exhibit little within-cluster scatter, then knowledge of an object's category permits a detailed inference about that object's features. Also, object categorization becomes less sensitive to measurement noise when categories are well separated in feature space; the inference of an object's category from observable features becomes more reliable. However, if the degradation of an object's description is caused by the omission of features as opposed to being caused by noisy measurements, then separated clusters do not insure reliable categorization. Separation in fea-

ture space is not equivalent to *redundancy* in feature space.[4] As discussed in chapter 2, categorization for recognition requires being able to determine an object's category from only partial information. Thus, while the optimization criteria used for cluster analysis are related to the proposed goals of categorization and recognition, they are inadequate for producing a suitable set of categories.

Given a clustering criteria, the problem of finding the best set of classes is well defined. Because there is only a finite number of data points n, there is only a finite number of partitions of the data into c classes; clustering reduces to finding the best partition. Unfortunately, the number of possible partitions is on the order of $c^n/c!$ (when $n \gg c$, see Rota [1964]), making exhaustive search impossible even for a relatively small number of data points. Therefore, the second component of the optimization procedure is the search algorithm used to find good clusters.

One approach is to use a pruned complete search, a form of branch-and-bound [Winston, 1977]. Even this method, however, quickly becomes combinatorially intractable. (Hand [1981] provides an example with $n = 20$, $c = 3$, where the pruning reduced the search by a factor of 1000, but still left almost one million partitions to be considered.) A more common method of search is that of gradient descent, where objects are incrementally transferred from one cluster to another to improve the clustering criterion. In the k-means method, clusters are modified by transferring each point to the cluster whose mean is closest to that point. However, such a method is sensitive to the initial hypothesis and, as with all gradient descent algorithms, may terminate at a local minimum. One radical approach to search is to try random partitions in an effort to find one of the best m partitions by testing a set of M partitions. [Fortier and Solomon, 1966]. Simple probability theory can determine how large M must be in order to be likely to discover one of the m best partitions. The difficulty with this approach is that M grows too quickly with n for some fixed probability of success. In chapter 5 we will develop a similar strategy for recovering classes of objects, but will apply the random search to only small subsets of the n samples. By restricting the the random search to small sets, we can maintain a high probability of success without testing arbitrarily large numbers of partitions.

[4] In chapter 4 we will provide a formal definition for redundancy. For now, let us assume redundancy measures how easily one can categorize an object given only a partial description of that object.

As a final comment we note that the use of optimization methods usually requires the a priori knowledge of the number of clusters present; often, such a priori information is not available. One solution to this problem is to augment the search procedure with the ability to split or merge categories at "appropriate" times; such a capability also allows optimization methods to cope with the addition of new data points. The well known ISODATA program of Ball and Hall [1967] provides such a mechanism in a clustering program which uses the trace of **W** presented above as the optimization criteria. If the sum of the squared distances between the data points and a the mean of any cluster becomes greater than some user specified threshold, then the cluster is split into two clusters. Likewise, pairs of clusters are merged if their means are separated by a distance less than some other user specified threshold. This method is successful for limited domains where such thresholds can be specified. However, a more consistent approach to cluster splitting and merging is to do so whenever such a change will improve the criteria measure. Such a procedure is possible only if the clustering criteria is not biased towards having many or few clusters. For example, the sum of squared distances is always reduced by splitting a cluster and thus would bias the procedure to find many (in fact n) clusters. Because the measure of the quality of a categorization we will develop in chapter 4 is not biased to having many or few clusters, we will be able to split and merge clusters according to the improvement of the clustering measure.

3.2.4 Cluster validity

An alternative to adding a cluster formation and deletion ability to optimization methods is to simply execute the same optimization procedure for a range of c, and then to compare the results. However, to select one c over another requires being able to assess the *validity* of a clustering of some data. Likewise, when generating a taxonomy with a hierarchical method, one is guaranteed that there exists a clustering of the data into c classes for all c, $1 \leq c \leq n$. To determine which of these descriptions represents "structure" in the data requires some method of determining whether a particular clustering is an arbitrary grouping of the data imposed by the algorithm, or a grouping robustly determined by the data themselves. Unfortunately, few methods for answering this question exist, and most of these are weak.

One formal method of assessing cluster validity is based on statistical

assumptions about the distribution of the data. As an example, consider an optimization method which seeks to minimize the sum of the squared within-cluster distances. Because $c+1$ clusters will always fit the data better than c clusters, we cannot use the absolute measure to determine which clustering is to be preferred. However, suppose we assume that the underlying data are sampled from c normally distributed classes. Then, one can derive an expected value for how much the clustering criteria would improve by using $c+1$ clusters instead of c. (For details of such a derivation see Duda and Hart [1973].) By comparing this value to the actual improvement obtained by splitting the c clusters into $c+1$ clusters, one can determine the validity of the new cluster. This method is only applicable when some underlying distribution of the data can be assumed and thus has limited applicability to domains where one is attempting to discover the structure of data. Because cluster analysis is usually used as a tool for such discovery, statistical measures of validity are highly suspect.

A simpler, intuitive approach to the validity problem may be referred to as "leave some out" methods [Hartigan, 1975]. In these methods either some of the data points or some of the dimensions used to describe the samples are omitted while executing the clustering procedure. After a set of clusters is generated (or, in hierarchical methods, selected from the taxonomy) the additional data points or dimensions are checked for consistency. A standard, though weak, method of checking is to test the statistical distribution of the additional sample points or dimensions. For example, the distribution of values along a previously omitted dimension would be checked for statistically significant differences between clusters. If such a difference is found, then the belief that the discovered clusters reflect structure in the data is strengthened. The weakness of this method is that not finding a significant difference only determines that some particular dimension is not constrained within the clusters discovered. Assuming a sufficient quantity of constrained dimensions, one would test other omitted dimensions, hoping to find constrained dimensions that supported the clustering discovered. Because of the qualitative nature of these methods, little formal analysis is possible.

The problem of whether an unconstrained dimension disconfirms the belief that a particular clustering is valid brings to light a fundamental shortcoming of cluster analysis: there is no a priori criteria for success. Let us assume that we have arbitrarily fast computing machinery and that we select the optimal partition of some data according to a particular clustering

criteria. It makes no sense to ask whether these clusters are "valid" classes, because *by definition* the groups which minimize the metric are the right groups. Therefore, if one wants to be able to say that the recovered classes are "valid" for some task, then it must be the case that the criteria used directly measures validity of the classes for the task at hand. In chapters 1 and 2 we defined the categorization task to be that of creating a set of categories that permitted robust categorization and the reliable inference of an object's properties from its category. Thus, if we are to create such a set of categories, will need to create a metric which directly measures these aspects of a categorization.

3.2.5 Clustering vs. classification

It is important to note the difference between cluster analysis as described above and *pattern classification* [Hand, 1981]. The term "classification" usually refers to the problem of deciding to which of a known set of categories a novel object belongs. Most of the pattern recognition and classification literature does not address the problem of discovering categories in a population of objects. It is assumed that a data analyst will provide a representative set of known instances; the problem of classification is to determine a measure or procedure by which new objects can be correctly classified.

However, one aspect of classification theory does relate to the problem of discovering structure within data. Often, the goal of a classification program is build a *decision tree* that provides an algorithmic decision sequence that will correctly classify new objects [Quinlan, 1986; Breiman, et. al. 1984]. In constructing such trees, a trade-off exists between the mis-classification rate and the total complexity of the decision function, often measured by the number of nodes in the decision tree. Breiman et. al. [1986] suggest a pruning mechanism that combines the two criteria using a free parameter α. This combination of opposing criteria is similar to that proposed by Tversky [1977] for determining basic level categories and is thus subject to the same criticism: the complexity of the description — for Breiman, et. al. the number of nodes, for Tversky the number of categories — should not be confused with the utility of a set of categories. However, the principle of trading ease of category inference for a more powerful set of categories is important and will be central to the theory developed in this thesis.

3.2.6 Summary of cluster analysis

Let us summarize the aspects of cluster analysis relevant to the task of object categorization. Three serious deficiencies in cluster analysis techniques were identified. First, the use of a distance metric which requires constraint in all dimensions and is applied uniformly throughout feature space is inappropriate for natural object categorization. Different processes in the world constrain different properties of objects and one must expect that each class of objects will have unconstrained dimensions. Second, methods of category formation that require the entire set of data be available initially (such as hierarchical methods) are not applicable in the natural world where new objects are often encountered. Third, optimization criteria are only useful if they directly measure the utility of the categories for a particular task. Criteria based only on distance in feature space cannot guarantee the formation of categories which permit the inferences required for the recognition task.

However, two positive aspects of cluster analysis were also noted. First, the dendrogram formed by hierarchical methods provides a method for testing the stability of a clustering with respect to the distance between clusters. We argued that it might be possible to test the validity of a categorization if the "distance" axis was sensitive to different processes involved in the creation of the data points (objects). Second, the tight hyper-sphere categories preferred by the criteria based upon scatter-matrices begin to satisfy the goals of a categorization established in chapter 2: the reliable inference of an object's properties from its category, and the reliable inference of an objects category from its properties. Better categories can be chosen only if the clustering criteria directly measure how well the categories support these goals.

3.3 Machine learning

The last field of research we must consider is that of machine learning. Machine learning is concerned with the issues involved in constructing a machine (program) that can discover structure in the world by examining specific instances. Whether the problem is to "discover" the laws of thermodynamics by "observing" experiments [Langley, Bradshaw, and Simon, 1983] or to learn the rules integration by being shown examples [Mitchell, 1983], the basic learning step requires *induction*: the formation of a general conclusion

based on evidence from particular examples. Unlike deduction programs which derive conclusions known to be true, induction programs are constructed such that the conclusions they derive are *likely* to be true.

For example, the BACON program [Langley, Bradshaw, and Simon, 1983] is able to discover scientific laws such as $F = m\,a$, $V = IR$, and $PV = nRT$. The reason BACON is successful in these cases is that the program explicitly seeks relations formed by simple additive and multiplicative manipulations. That is, embedded within the program is the belief (on the part of the programmer) that if relations of this form adequately describe the data, then these relations are the correct natural laws. Furthermore, there is the belief that laws of this form exist, thereby justifying a search for these relations.

3.3.1 Conceptual clustering

One focus of machine learning which is relevant to the task of categorization is in the area of *conceptual clustering* [Michalski, 1980; Michalski and Stepp, 1983a,b], a paradigm similar to the cluster analysis methodology presented above. As in cluster analysis, the task at hand is to categorize a set of data points into "good" classes. However, in conceptual clustering the notion of "good" is not (solely) based upon a distance metric, but also on an a priori belief as to what types of cluster descriptions are "natural." Similar to the discovery program BACON which makes an assumption about the *form* of a natural law, conceptual clustering programs make an assumption about the form that descriptions of natural clusters should have. We shall need to relate the particular beliefs about the desired form for descriptions of natural classes to the goals of categorization and the principle of natural modes presented in chapter 2.

As an example of conceptual clustering, let us consider the work of Michalski and Stepp [1983a,b]. In their system — CLUSTER/2 — data points are represented as feature vectors, but the the dimensions are not necessarily ordinal. Typical features would be "shape" or "color" which could take values such as "red" or "round," respectively. Convex subsets[5] of data points are described by conjunctive combinations of internally dis-

[5]Convex is not exactly the correct description since the nominal features (e.g. "color") are not metric. However, if they were, and they were arranged (just for this conjunction) such that the internal disjunctions (e.g. red $\vee$ blue) were sequential, then the sets would be convex.

junctive *feature selectors*; these combinations are referred to as *conjunctive complexes*. For example,

$$[\text{shape} = \text{round}][\text{color} = \text{red} \vee \text{blue}][\text{size} \geq \text{medium}]$$

would describe the set of all red or blue round things that are at least size medium. Arbitrary clusters can then be represented by the union of such conjunctive complexes; these unions, which are made as simple as possible by eliminating any redundant complexes, are referred to as *stars*. When a set of clusters is finally chosen, the stars may be used as conceptual descriptions of the clusters. Unlike standard cluster analysis, conceptual clustering produces a description — claimed to be conceptual — of the recovered classes.

Michalski and Stepp describe a procedure for clustering similar to the k-means method of cluster analysis described in the previous section. An initial set of c "seed" data points are chosen and "good" clusters described by the unions of conjunctive complexes are built around those seeds. Then, an iteration loop is executed in an attempt to select new seeds that yield better clusters. The details of how seeds are selected, and of how clusters are constructed are not important for relating this work to the problem of categorizing objects for recognition. Of interest are the criteria used to judge the quality of a clustering, and how those criteria relate to the proposed goals of categorization and the principle of natural modes.

Michalski and Stepp describe four component criteria relevant to the present discussion. Each represents a different, intuitively desirable property for "good" clusters. The first two — *commonality* and *disjointness* — resemble the scatter matrices of cluster analysis. Commonality refers to the number of properties shared by data points within a cluster; if sharing of properties is used to define a distance metric, then commonality resembles the inverse of the within-cluster scatter. Likewise, disjointness measures the degree of separation — how little they overlap — between each pair of complexes taken from different stars; this measure is analogous to the between-cluster scatter. As previously mentioned, clustering criteria based upon these scatter matrices favor categories that are tight hyper-spheres in feature space. Also, as discussed, such categories begin to satisfy the criteria of categorization proposed in chapter 2.

The next component of the clustering criteria reflects an assumption about the goal of categorization. *Discriminability*[6] measures the degree of

[6]Michalski and Stepp describe different versions of discriminability in two presentations

ease in determining the cluster to which an object belongs given only a partial description of the object. As a clustering becomes more discriminable, less information is required (on average) to identify an object's category. This criteria corresponds to one of the goals of categorization outlined in chapter 2: reliable categorization when provided with partial information.

The final element of the clustering criteria of Michalski and Stepp is *simplicity*, and it is an assumption about what constitutes a "meaningful" category in the world. Simplicity is defined as the negative of the complexity, which is simply the total number of feature selectors in all the cluster descriptions. This criterion reflects the assumption that the most meaningful categories are those that can be described by a small number of properties. Let us consider the validity of the simplicity criterion in light of the principle of natural modes In one respect, simplicity is consistent with a modal world: if natural classes are highly clustered in the space of important environmental properties, then only a small number of these properties need be described to classify an object. However, when posed in this manner, this criterion is equivalent to the discriminability criterion above. The more fundamental meaning of simplicity is that the clusters are *defined* by a small number of properties; this is the view of simplicity intended by Michalski and Stepp, as they refer to the conceptual description of the clusters as the "meaning" of the classes. In this light, simplicity is at odds with the principle of natural modes, which posits the existence of highly structured, complex classes. These categories are discriminable because their *complex* structures are highly dissimilar; complex environmental pressures cause objects' configurations to be different from one another in a large number of dimensions. Thus, simplicity — an intuitively appealing criterion — cannot be regarded as consistent with the goal of categorizing objects according to their natural modes.

---

of their clustering procedure [Michalski and Stepp, 1983a,b]. The *discrimination index* is defined to be the number of dimensions that singly distinguish all of the clusters — they take on a different value for each cluster. *Dimensionality reduction* is defined to be the negative of the number of dimensions required to uniquely identify the cluster to which an object belongs; the negative value is used so that the value increases as a clustering becomes more discriminable. If the discrimination index is greater than zero (at least one dimension singly distinguishes all of the clusters), then the dimensionality reduction must be -1. We can define *discriminability* to be the sum of these two values: the greater the value, the less restricted is the information that will uniquely determine an object's category.

In summary, conceptual clustering represents an improvement over standard cluster analysis. Besides the the advertised extension of providing a description of the created clusters, conceptual clustering utilizes criteria that consider the goals of the observer — discriminability improves reliability — and an a priori belief about the structure of natural classes — simple classes are preferred. However, we have argued that the assumption that simple descriptions are the right descriptions is not valid for the task of categorizing objects in the natural world; natural objects are highly constrained and thus complex in structure. Furthermore, conceptual clustering faces the same category validity problem as cluster analysis. The categories recovered are those which optimize the particular set of criteria chosen; the criteria were not chosen according to some task requirement. Thus it is difficult to assess the utility of the recovered classes for a specified task such as that proposed in chapter 2: the reliable inference of an object's properties from its category.

We have not presented the method used by Michalski and Stepp to find possible clusters (they use a form of a bounded best-first search) as it resembles search techniques used in standard cluster analysis. The procedure is iterative and not well suited to a system which must dynamically generate categories as new data are observed. Also, the computational expense of forming these good, but certainly not optimal, clusters is almost prohibitive.[7]

3.3.2 Explanation-based learning

We stated that the criteria of simplicity used by Michalski and Stepp [1983a,b] reflected an assumption about the structure of categories in the world. As with all similarity-based methods, the vocabulary on which the syntactic operations are performed (operations such as measuring the complexity of a cluster by counting the number of feature selectors used) implicitly embodies a *theory* about the world. As demonstrated in the proof of the Ugly Duckling Theorem in chapter 2, a different set of predicates can cause previously "simple" categories to become "complex." Unfortunately, the theory embedded in similarity based techniques always remains implicit in the vocabulary. Thus it is difficult (if not impossible) to improve one's theory through experience, and it is difficult to evaluate the correctness of a theory except by

[7]For a critique of the conceptual clustering work see [Dale, 1985].

the actual execution of the similarity-based algorithm.

Recently, a new form of machine learning — referred to as *explanation-based* — has been developed in an attempt to incorporate an explicit theory about a domain into the learning process. For example, the LEX program of Mitchell [1983] uses a priori information about mathematical relations to learn the rules of symbolic integration from examples provided by a teacher. Instead of just using syntactic rules for comparing one formula to another, the program uses its knowledge about mathematical functions to form its generalizations. For example, part of its theory includes the fact that `sin` and `cos` are both trigonometric functions. Therefore, when it is told that the integration of $x \sin x$ can be accomplished by `integration-by-parts`, the program hypothesizes a generalized rule that states $x \operatorname{trig} x$ can be integrated by "integration-by-parts." This rule is maintained unless a counter example is provided by the teacher.

Certainly, an explanation-based approach to categorization would be a more powerful technique than simple similarity-based methods [DeJong, 1986];[8] at present we are unaware of any such attempts. Such an approach would require an underlying theory of physics of natural objects. The program would have to know what types of equivalence classes can be created by different object processes. Evidence for such a strategy existing in organisms may be found in the work of Cerella [1979] in which pigeons were able to form a natural category for "white-oak-leaf" from the presentation of just one instance. The pigeons must have an underlying theory that determines which aspects of the physical structure of the leaf are likely to be important in determining its natural class. In chapter 7 we will consider some possible extensions to the work presented in this thesis; the most interesting of these incorporates knowledge of physical processes into the mechanism for recovering natural object categories.

[8]Though see Liebowitz [1986] for a discussion of the relationship between similarity-based and explanation-based methods.

Chapter 4

Evaluation of Natural Categories

In chapter 2 we argued that the goal of the observer is to form object categories that permit the reliable inference of unobserved properties from observed properties; we claimed that to achieve this goal the observer should categorize objects according to their natural modes. To accomplish this task, the observer must be provided with two separate capabilities. First, he must be able to identify when a set of categories corresponds to a set of natural clusters. This ability requires that the observer be given criteria with which to evaluate a particular categorization. The second capability required is that of being able to make "good guesses." Chapter 3 included a section on the search strategies used by optimization methods of cluster analysis; such a search strategy is necessary because of the enormous number of possible partitionings of a set of objects. Likewise, to discover "the correct set" of categories, the observer must consider that particular set as a possible candidate. In this chapter we develop a measure of the extent to which a categorization allows the observer to make inferences about the properties of objects. We defer the problem of generating suitable hypotheses until the following chapter.

We proceed by first considering only the goals of the observer, and deriving an evaluation function which measures how well a particular categorization of objects supports these goals. We then describe the behavior of this measure in both a structured (natural modes) and unstructured world. Finally, by means of an example drawn from the natural domain of leaves,

we demonstrate the the ability of the measure to distinguish between natural and arbitrary categorizations.

4.1 Objects, Classes, and Categories

First let us define some necessary terminology. We assume there exists a fixed set of *objects,* $\{\theta_1, \theta_2, \ldots, \theta_n\}$; Θ denotes the set of all possible objects. As mentioned in chapter 1, we will not provide a definition for "object," though at the conclusion of the thesis we will consider using the construct of a category to define criteria for being an object. A *categorization*, $\mathcal{Z}$, is simply a partitioning of this set of objects, with each equivalence class defined by the partition being referred to as a *category.* Notice, that in this terminology (and for the remainder of this thesis) categories and categorizations are mental constructs, hypotheses and conclusions made by the observer. In section 4.3.1 we will develop a formal notation for deriving expressions involving categories and categorizations. The goal of the observer is to create a categorization of objects that support the goals of inference established in chapter 2.

When we need to refer to the structure of objects in the world, we will refer to object *classes.* Thus the principle of natural modes states that objects in the world are divided into natural classes; these classes are produced by the natural object processes discussed in section 2.4. Because the discussion of this chapter will focus on the evaluation of the observer's proposed categorizations, we will not provide a more extensive definition of classes; for a more formal discussion about classes see Bobick and Richards [1986].

4.2 Levels of Categorization

We begin our development of a measure of how well a categorization supports the goals of the observer by considering object *taxonomies,* such as that pictured in Figure 4.1. (As is often the case, trees in computers grow upside-down: the root node THING is at the top; the leaves, e.g. "Fido", at the bottom.) Each non-terminal node represents a category composed of the union of the categories below it. The terminal nodes — the "leaves" of the tree — are categories containing exactly one object. Given a set of objects, one may create a large number of taxonomies. For the purposes of

developing a measure of the utility of a categorization we will assume that some particular taxonomy has been provided.

Notice that the set of categories at any level of the taxonomy constitutes a partitioning of the set of objects and is thus a legitimate categorization. Suppose our task is to select the level which best allows the observer to satisfy his goals of making reliable predictions about unobserved (and observed) properties.[1] Let us assume that the observer will make these predictions based upon the category to which he assigns some object and the properties of other objects known to be of that category. Therefore, to select the best level of the taxonomy, we have to consider how the depth of the categorization affects the ability of the observer to correctly categorize an object and the ability to make predictions about an object once its category is known.

For the remainder of this chapter, we will be considering only observed properties, since we assume that those properties are the only ones available to the observer for evaluation of a categorization. As discussed in section 2.4.2 the unobserved properties should behave similarly to the observed properties. Therefore, we assume that a categorization that provides good performance in terms of predicting observed properties, and that allows reliable categorization based on those properties, will also be good for predicting unobserved properties.

4.2.1 Minimizing property uncertainty

First, consider moving down the tree from the root towards the leaves, moving from THING to "Fido" (Figure 4.2). In doing so, the categories become more specialized: knowledge that an object belongs to the category provides more information about the object. For example, knowing that some object is a dog allows the observer to predict many more properties (e.g. *has teeth*, *has hair*, *has legs*) than if he only knew the object was an animal. At the extreme depth of categorization, each category contains only one object. Let us assume the observer knows everything there is to know about each

[1] We should point out that it is somewhat artificial to require selecting some particular *level*. This presupposes that the same level is the best level across the entire taxonomy tree. A more appropriate task would be to pick the best set of spanning nodes, since the "best" level in one part of the tree may be lower than that of another part. For the principles to be developed here, however, considering a fixed level of categorization is sufficient.

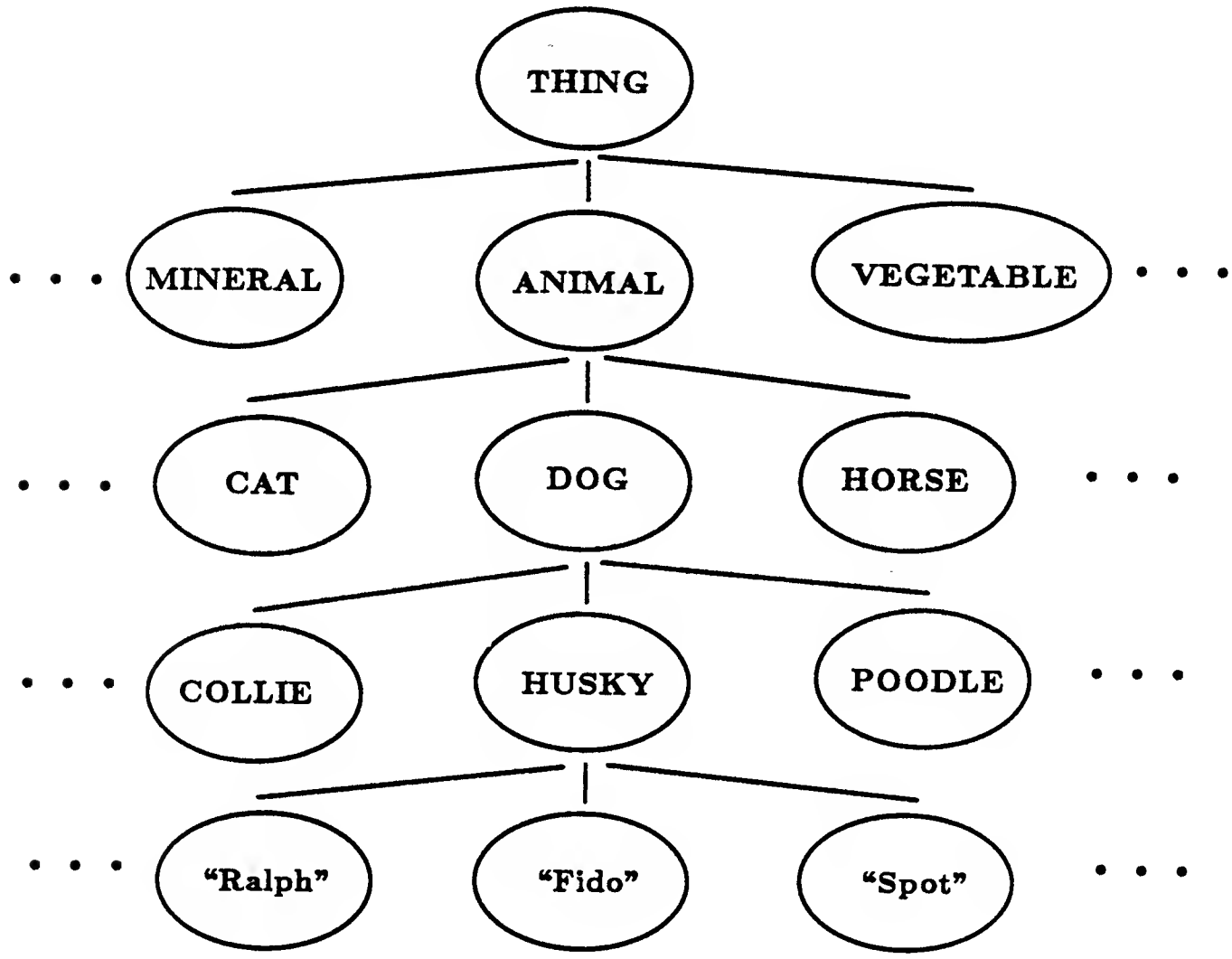


Figure 4.1: A complete taxonomy of objects. The question to be considered is what is the appropriate level of categorization given the goals of recognition It should be noted that number of possible taxonomies is enormous. For example, if there are 128 objects, then the number of balanced binary taxonomies is approximately 10^{55}.

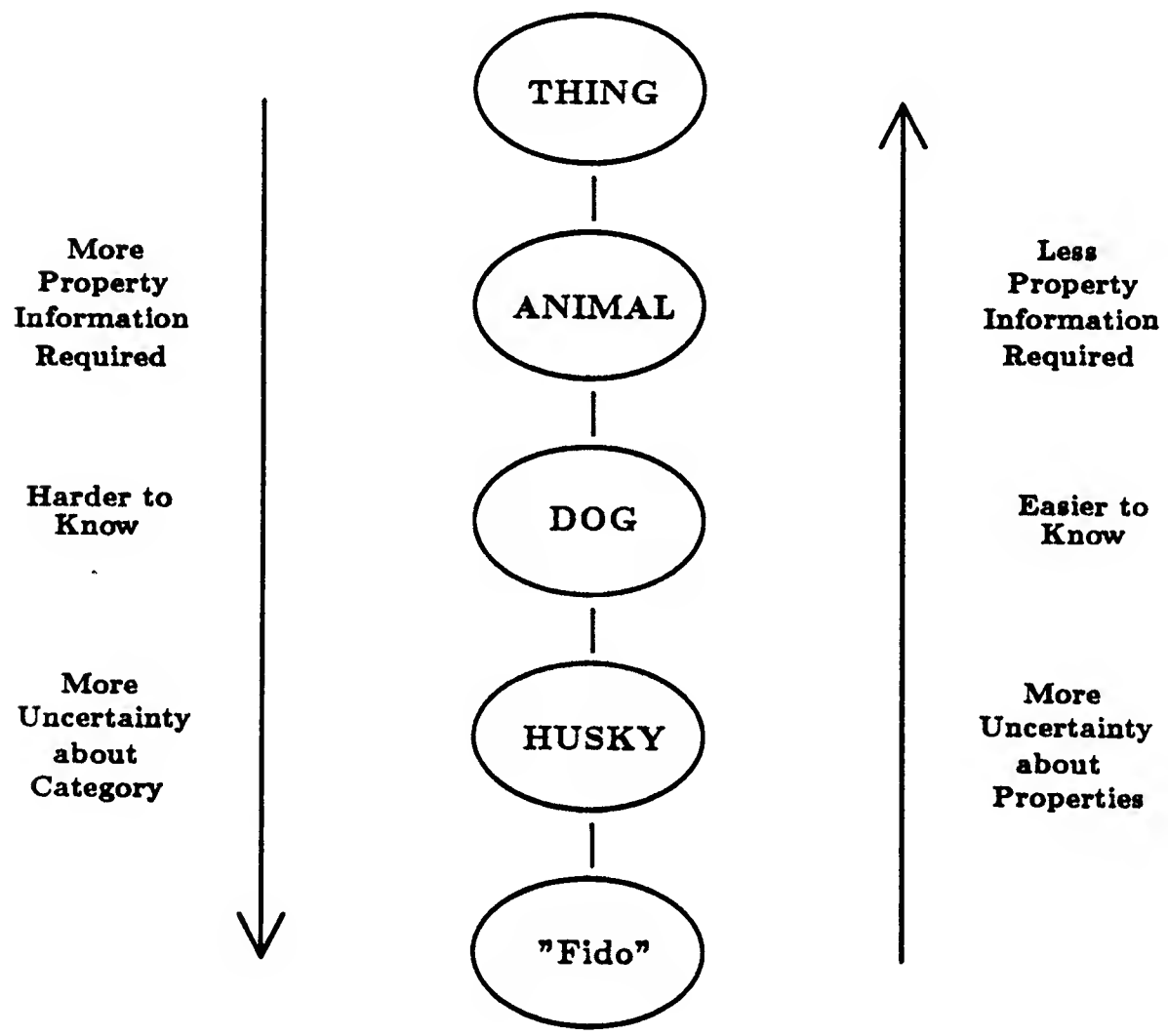


Figure 4.2: A single path from THING to "Fido" in some proposed taxonomy tree. As the depth of categorization increases, more predictions about unobserved (and observed) properties are possible; however, classification of an object becomes more difficult and less reliable, and the ability to categorize novel objects will degrade.

instance (e.g. "Fido"). Then, at this finest level of categorization, knowledge of the category to which an object belongs allows complete prediction of its observed and unobserved properties.

We can describe the process of increasing the depth of categorization as minimizing *property uncertainty*, which we denote as $U_P(\mathcal{Z})$. This uncertainty decreases as objects in a category becomes more "similar" to each other. Thus property uncertainty measures the inhomogeneity of each category and expresses the difficulty the observer has in attaining his goal of being able to predict properties of an object once its category is known. Later, we will propose an explicit measure for U_P which is claimed to be appropriate for perception. For now we simply note that U_P decreases as we move down the taxonomy hierarchy.

There is, however, a price to be paid for increased categorization depth and the reduction of property uncertainty. As categories become smaller and more refined, the differences between categories becomes smaller, making the task of categorization more difficult and less reliable. For example, to determine that an object is a Siberian Husky generally requires more property information than to determine that it is a dog. Furthermore, the categorization of novel objects becomes less reliable since different categories are now more similar to each other; deciding whether some new object is a Husky or a German Shepherd is more difficult than deciding whether it is a Dog or a Horse. Thus, increasing the depth of categorization facilitates some goals of the observer while hindering others.

4.2.2 Minimizing category uncertainty

Now, let us consider climbing the taxonomy tree, with the categorizations becoming coarser as we move from the finest categories to the root node THING. Now the categories become more general: knowledge of the category to which an object belongs provides less information about the properties of the object as we get closer to the root node. At the extreme, where there is only the one category THING, knowledge of an object's category permits almost no predictions about any of its properties. Therefore, decreasing the depth of categorization decreases the ability of the observer to satisfy his goal of being able to make important predictions about objects based upon their categorization.

As to be expected, the sacrifice of the ability to make predictions about

the properties of objects is accompanied by a compensatory increase in the ease of categorizing an object. In general, less property information is required to know that an object is a dog than to know that it is a Siberian Husky.[2] In the case of the depth zero categorization, where the only category is THING, minimal information is required to make the correct classification.[3] Likewise, the ability to categorize novel objects also improves with decreasing categorization depth since the categories become more encompassing. Climbing up the taxonomy tree reduces the *category uncertainty* which we denote as $U_C(\mathcal{Z})$. As with increasing categorization depth, decreasing depth facilitates some of the recognition goals of the observer and hinders others.

To further refine our definition of U_C, we need to make the categorization process more explicit. Let us assume that the process of categorizing an object is performed by looking at the current categorization of objects and finding the category whose objects "match best" — an operation which we will currently leave undefined — the object in question in their observed properties. If given a complete description of an object, and if that object matches only objects in one category, then there is no uncertainty in the categorization process. However, in perception it is often the case that many of the potentially observable properties are not provided in an object's description or that an object matches no object in the current categorization or that it matches objects in several categories. Therefore let us loosely define the category uncertainty as the uncertainty of the category of an object given *some* of its observed properties. This definition also accounts for ob-

[2]Note there may exist some unique identifying property which will indicate membership in some low level category. For example, if one knows that an object has one blue eye and one brown eye, then there is a high probability that the object is a Siberian Husky. Thus, for that particular property, identifying an object as a dog is no easier than identifying it as some particular type of dog. However, two points help eliminate this concern. First, *by definition,* any property which helps to categorize an object as a Siberian Husky also helps to categorize that object as a dog. Therefore determining an object is a dog can be no more difficult that determining it is a Husky. Second, if we assume the difficulty of categorization is measured not only by the number of properties required to categorize an object but also by how restricted those properties must be, then the existence of some unique identifying feature does not make the Husky categorization easier. Later in this chapter we define a formal measure of the uncertainty in categorizing an object that is consistent with this assumption.

[3]We say "minimal" information as opposed to none because some information might be required just to know something is a "Thing." For example, is sand in a sandbox a thing? This problem cannot be resolved with defining what constitutes an object.

jects that don't match any previous object since they presumably do match other objects in some of their properties. In later sections where we derive a particular evaluation function, we will make more precise the idea of some observed properties.

4.2.3 Uncertainty of a categorization

In the previous two sections we noted that as a categorization becomes either finer or coarser, some of the goals of the observer are made more difficult to achieve while others are made easier. Therefore, if the observer requires some degree of success in all his goals, then the appropriate level of categorization must lie somewhere between the two extreme granularities of categorization. However, the question as to what level is appropriate can not be answered until the desired relative achievement of the conflicting goals of the observer is specified.

For example, consider an organism that has both simple perceptual needs — the properties he needs to extract about objects are few and quite crude — and a primitive sensory apparatus — the extraction of complicated information is quite difficult and time consuming. Such an organism would desire a set of categories relatively near to the top of an object taxonomy. Choosing such a set corresponds to sacrificing the ability to make precise predictions about the properties of objects in exchange for reliable and time effective categorization. Inversely, an organism with great perceptual demands and refined sensory mechanisms (e.g. primates) would make the opposite choice: a set of categories that required encoding more sensory information but afforded more precise predictions.

Let us propose a categorization evaluation function that makes explicit the trade-off between these two conflicting goals of the observer. We assume we have a candidate categorization — a partitioning of the set of objects into a set of categories — and that our task is to evaluate how well the categorization satisfies the goals of the observer. We require an evaluation function that combines U_P and U_C in such a manner as to make explicit the trade-off between the two uncertainties. Let us introduce the parameter λ to represent that trade-off, and let $U(U_P, U_C, \lambda)$ be the *total uncertainty* of a categorization, where $0 \leq \lambda \leq 1$. We will view U as as measure of poorness of a categorization; the less total uncertainty a categorization has the more it is to be preferred. λ is to be interpreted as a relative weight

between being able to infer an object's properties from it's category and being able to infer an object's category from its properties. When $\lambda = 0$ only the ability to infer properties is considered; thus the best categorization is that which is the finest. Likewise when $\lambda = 1$, only the ability to infer the category is important; in this case the coarsest categorization is preferred. The important questions that arise are what are the preferred categorizations as λ takes on intermediate values and how does the setting of λ interact with the actual classes present in the world. We will have to postpone the discussion of these issues until after we derive suitable measures for U_P and U_C.

4.3 Measuring Uncertainty

4.3.1 Property based representation

The observer does not directly categorize the objects in the world. Rather, he can only operate on a *representation* of those objects. We define a representation to be a mapping from the set of all possible objects, Θ, to some finite set Θ^*.[4] Note that even though we required that each object θ_i be a member of only one category (the categorization is a partition in the mathematical sense) two distinct objects may have the same description in the representation used by the observer. The representation of object θ_i may be identical to the representation of object θ_j, but since it is a different object, it is permitted to be in a different category. Of course, if one is proposing that the categories of some categorization correspond to natural mode classes, then this situation would either be a violation of the Principle of Natural Modes or simply a representation insensitive to the differences between classes.[5] However, as a potential categorization it is certainly permissible. Furthermore, a single category may have many objects with the same description, which corresponds to the situation where the representation does not discriminate between two objects assigned to the same category.[6]

---

[4]The finite restriction is included to agree with the intuition that there is some limit to information encoded by the observer.

[5]In chapter 5 we will further consider the competence of a representation.

[6]A problem with allowing distinct objects to have identical descriptions is that it becomes impossible to distinguish between the case of two different objects being so similar that they map to the same point in the representation space and the case of two instances of

For our derivation of the quantities U_P and U_C we will utilize a property based representation. Though commonly referred to as *feature space* representation [Duda and Hart, 1973], we prefer the term property description to emphasize the fact that these properties are of the objects themselves, not of an image or some other sensory representation. The term "feature" will be used, but to refer to a predicate defined on objects and computable from sensory information. We should note that the form of the representation is not critical to the qualitative results derived about the evaluation function. If one prefers some other representational form, for example the volumetric primitive approach of generalized cylinders, then such a representation may be used as long as a method for computing U_P and U_C is also specified.

The terminology of our property based representation is defined as follows: the term *feature* refers to a function or predicate computed about an object; the term *value*, to the value taken by a feature; the term *property*, to the valued feature. For example, "length" is a feature, "6 ft." is a value, and "having length 6 ft." is a property. Each feature, f_i, $1 \leq i \leq m$, has an associated set of values $\{v_{i1}, v_{i2}, \ldots, v_{i\eta}\}$ referred to as the range of the feature. We require that the range be a finite set but the cardinality of the range can vary from one feature to the next. $\mathbf{F}$ denotes the set of features $\{f_1, f_2, \ldots, f_m\}$ Using these features, each object θ is represented by an m-dimensional property vector $\mathbf{P} = (v_{1\alpha}, v_{2\beta}, \ldots, v_{m\gamma})$ where v_{ij} is the j^{th} value of the range of the i^{th} feature.

As defined at the start of this section, a categorization is a partitioning of the population of objects, with each equivalence class defined by the partition being referred to as a category. The symbol $\mathcal{Z}$ will continue to be used to represent some possible categorization; often, however, the operations being discussed will only be meaningful with respect to some categorization and the explicit use of $\mathcal{Z}$ will be omitted. In the sections that follow we let c be the number of categories in a categorization, and let C_i be the i^{th} category. Also, we need a category function, Ψ which maps an object onto its category in the current categorization: $\Psi(\theta_k)$ is the category to which the the object θ_k belongs. We denote $\Psi(\theta_k)$ as ψ_k. The size of a category is expressed by $\|C_i\|$ or by $\|\psi_k\|$ depending on whether referring to the i^{th} category or the category to which object θ_k belongs.

Finally, when we need to refer to the structure of objects in the world,

---

the same object. For now, we assume that somehow we know that each object is distinct.

we will need to refer to the natural classes present. Recall that a class is distinguished from a category in that a class represents structure in the physical world, where as a category is part of a categorization proposed by the observer. We will use the symbol Ω_j to represent the j^{th} class.

4.3.2 Information theory and entropy

In our discussion about the ability to make inferences about the properties of objects, we have been using the term *uncertainty* without having provided a suitable definition. If we are to propose a measure of the utility of a categorization based upon uncertainty of inference, we must have a formal definition of uncertainty consistent with the representation of objects and categories.

In information theory, uncertainty is the amount of information which is unknown about some signal [McEliece, 1977]. It is measured in terms of the probabilities of the signal being in each of its possible states. For example if some signal A can be in one of two states, each with probability .5, and signal B can be in one of 4 states each with probability .25, then there is said to be more uncertainty about signal B, and signal B is said to convey more information. Shannon, in his original work on information theory [Shannon and Weaver, 1949], derived an information measure H based upon the *entropy* of a probability distribution:

$$H = -\sum_{i=1}^{m} p_i \log p_i \tag{4.1}$$

where $p_i \geq 0$, and $\sum_{i=0}^{m} p_i = 1$. One of the elegant results of that work was the demonstration that any measure of uncertainty must use a $p \log p$ formulation if it is to satisfy several desirable and intuitive constraints about information and communication. As such, entropy has become the standard means of measuring uncertainty [McEliece, 1977].

The question we need to consider is whether it is appropriate to consider the uncertainty in the perceptual process to be similar to uncertainty in the theory of communication. If so, then entropy is a natural measure in which to express uncertainty. Perhaps the simplest answer to this question is that *perception is communication.* We can view the perceptual process as communication between what is being observed and the observer. The channel consists of the sensory apparatus; the coded message, the sensory

input. It is the task of the observer to decode the actual message from sensory input. As such we claim that the traditional measure of uncertainty in communication theory is an appropriate measure of perceptual uncertainty.[7]

Also, the particular form of the uncertainty measure is not critical to the work described here. In fact, an implementation not reported in this document made use of a measure based on the probability of making an error if the observer made his best guess about an object's category. The results of that implementation were similar to those achieved with the entropy measure.

When deriving the uncertainty measures of the next sections it will be useful to keep in mind three properties of the entropy measure H that are consistent with one's intuition about measuring uncertainty. First, $H = 0$ if there is only one possible state, i.e. $p_i = 1$ and for all $j \neq i$, $p_j = 0$. Thus, when only one alternative exists (say, about the category of an object) the uncertainty measure equals zero. Second, for the case when all the probabilities are equal, $p_i = p_j$ for all i and j, H increases as the number of choices increases. The greater the number of alternatives, the greater the uncertainty. Finally, for a fixed number of alternatives m, H is a maximum when all of the probabilities are equal, and that maximum value of H is $\log m$. Uncertainty is the greatest when one has no reason to prefer one alternative over another.

4.3.3 Measuring U_P

In this section we will derive an entropy measure for the property uncertainty U_P. We proceed by assuming that the observer knows that an object belongs to some particular category C_i. The question we want to answer is how much uncertainty does he have about the object's properties?

There are two ways to think about the properties of objects. The first is to consider the property vector as a whole, and the uncertainty of the properties of an object is the uncertainty of the entire property vector. The second is to consider each component independently. To decide which way is appropriate for measuring U_P, we must consider the tasks of the observer for which the property information is useful.

[7]The recent work by Bennett, Hoffman and Prakash [1987] on "observer mechanics" supports the view of perception as an encoding (and projection) of the state of the world.

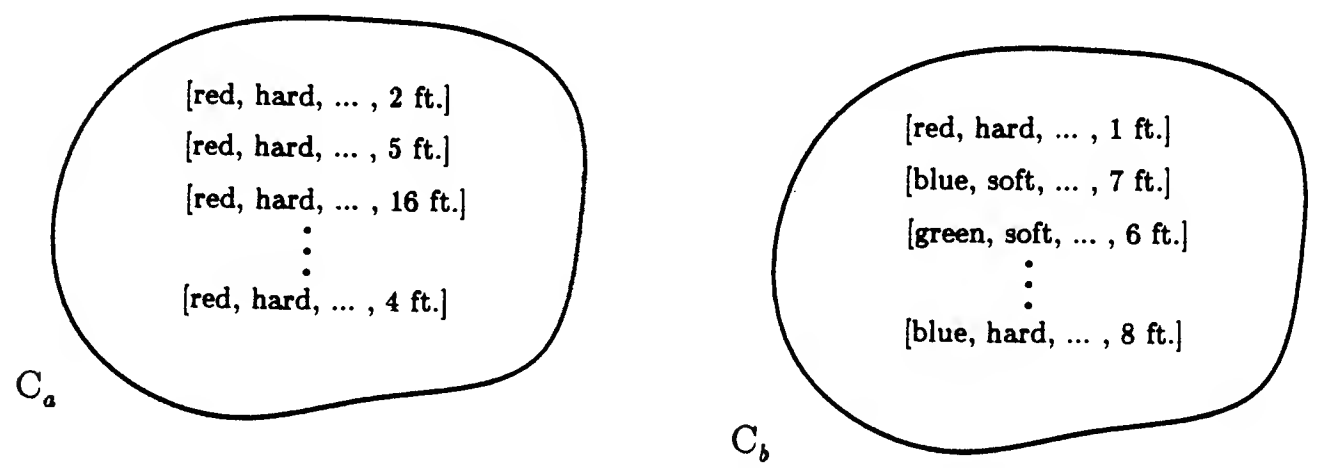


Figure 4.3: Two categories with their objects and associated property vectors. If the components of the property vector are considered independently then the uncertainty of C_b is greater than that of C_a; otherwise they are equal.

One such task is simply needing to know some particular property. For example, the fact that something is hard (therefore can be stepped on safely) or that something moves (therefore should be kept at a safe distance) are properties the observer might want to know directly. Another task, and one which may be critical for a reliable perceptual process, is the ability to make predictions about as yet unobserved but potentially observable properties; these predictions are necessary for the verification of the identity of an object. In both of these cases, it is the separate uncertainties of the components of the property vector that are important. Figure 4.3 illustrates this point. For the perceptual goals of the observer, knowing that some object is a member of category C_a provides him with more useful information than knowing that some object is a member of C_b. However, for the uncertainty of the properties of an object given its category to be greater for C_b than for C_a, then it must be the case that we consider the properties independently. If the property vectors are considered in their entirety, then the property uncertainties of C_a and C_b would be equal.

We now construct a measure of property uncertainty considering the

uncertainty of each of the features independently. First, we need to define the uncertainty of a feature in a category. To reduce the complexity of the notation, we define $H(D)$ to be the entropy of any finite probability distribution D:

$$H(D) = -\sum_{i=1}^{s} p_i \log_2 p_i \tag{4.2}$$

where $D = \{p_1, p_2, \ldots, p_s\}$, $p_i \geq 0$, and $\sum_{i=0}^{s} p_i = 1$. For the remainder of the thesis the base of the logarithm will be omitted from the expressions; we will always assume it to be 2.

Now let us define the distribution of a feature f_i in some category C_a. Let p_{ia}^j be the fraction of objects in C_a whose value for feature f_i that is equal to the j^{th} value in the range of f_i.[8] Then the $dist(f_i)$ in C_a is the set $\{p_{ia}^1, p_{ia}^2, \ldots p_{ia}^\eta\}$ where η is the number of values in the range of f_i. Using this distribution we define the uncertainty of feature f_i in category C_a to be $H(dist(f_i)$ in $C_a)$.

Having defined the uncertainty of a feature in a category we can define our property uncertainty of the category as the sum of the feature uncertainties:

$$U_{P\text{-}of\text{-}C}(C_a) = \sum_{f_i \in \mathbf{F}} H\left(dist(f_i) \text{ in } C_a\right) \tag{4.3}$$

The above equation provides a measure to answer the question of how much uncertainty about an object's properties remains once that object's category is known. To compute $U_P(\mathcal{Z})$ we must extend that measure to provide an evaluation of the property uncertainty over the entire categorization. Let n be the total number of objects, $n = \sum_i \|C_i\|$. Recalling that $\Psi(\theta_i)$ represents the category to which object θ_i belongs, we define U_P as the average of $U_{P\text{-}of\text{-}C}$ as summed over all the objects in the categorization $\mathcal{Z}$:

$$U_P(\mathcal{Z}) = \frac{1}{n} \sum_{\theta_i \text{ in } \mathcal{Z}} U_{P\text{-}of\text{-}C}(\Psi(\theta_i)) \tag{4.4}$$

Since $U_{P\text{-}of\text{-}C}$ is only a function of $\Psi(\theta_i)$ and not of θ_i itself, we can sum over the categories instead of the objects, weighting each category according to its size:

[8] We do not have to exclude the case where $p = 0$ because, by L'Hospital's rule, $\lim_{p \to 0} p \log p = 0$.

$$U_P(\mathcal{Z}) = \frac{1}{n} \sum_{C_i \in \mathcal{Z}} \|C_i\| \, U_{P\text{-}of\text{-}C}(C_i) \tag{4.5}$$

This second form is computationally less intensive and is the form used in the implementations discussed later in this chapter. We postpone discussion of how U_P behaves in ideal, noise, and real conditions until after we derive U_C and can apply a total uncertainty function to both artificial and real data. Later, we shall also discuss how U_P compares with some of the distance metrics discussed in chapter 3.

4.3.4 Measuring U_C

Having proposed a measure for U_P we must now provide a measure for U_C, the category uncertainty. In section 4.2.3, we stated that U_C was the uncertainty of the categorization of an object given *some* of the object's properties; we must now make that loose description precise.

To begin, let us assume that "some" of the observed properties means exactly what it says: we are given only some of the components of the property vector describing some object. This situation would arise if some of the (potentially) observable properties could not be recovered in the current sensing situation. Consider the uncertainty of categorization if we are given this incomplete description of the object and our task is to decide to which category that object belongs. To determine the correct category, the observer would check each category in turn, noticing whether there are objects whose property vector matches the components that are provided for the object in question. If only one category contains any objects that match, then there is no uncertainty of categorization. If, however, there is more than one category, we need some way of measuring the uncertainty as to which category the object belongs.[9]

We will design a measure of category uncertainty by assuming that the percentage of matches that a partial description of an object makes to a category is representative of the probability that the object actually belongs to that category. For example, suppose a given partial description of object

[9] We do not need to consider the case of an object not matching any of the objects in the categories. The uncertainty measure is designed for the evaluation of a categorization in which all objects of the population have been categorized. Thus every object is guaranteed to match at least one object, namely itself.

θ_k, matches 4 objects in category C_a, 12 objects in category C_b, and no objects in any remaining category. Then we say the probability that θ_k belongs in C_a is .25, and in C_b is .75. This suggests that we measure the uncertainty of categorization for an object with the entropy function H.

Let $\mathbf{F}'$ be some subset of the set of features $\mathbf{F}$. We define $\textsc{Match}(\theta_k, C_a, \mathbf{F}')$ to be the number of objects in category C_a whose property vectors match that of θ_k in the components contained in $\mathbf{F}'$. As such we define the match probability $p_M(\theta_k, C_a, \mathbf{F}')$ of θ_k in C_a on $\mathbf{F}'$:

$$p_M(\theta_k, C_a, \mathbf{F}') = \frac{\textsc{Match}(\theta_k, C_a, \mathbf{F}')}{\sum_{C_i} \textsc{Match}(\theta_k, C_i, \mathbf{F}')} \tag{4.6}$$

where the denominator is simply the sum of the matches over all the categories. Given the match probabilities, we define the *match distribution* of $(\theta_k, \mathbf{F}')$ to be the set of probabilities $\{p_M(\theta_k, C_1, \mathbf{F}'), p_M(\theta_k, C_2, \mathbf{F}'), \ldots, p_M(\theta_k, C_c, \mathbf{F}')\}$. Finally, we can define the category uncertainty for a given object with a given feature subset description:

$$U_{C\text{-}of\text{-}\theta}(\theta_i, \mathbf{F}') = H(\textit{match distribution} \text{of}\, (\theta_i, \mathbf{F}')) \tag{4.7}$$

If an object θ_i matches only objects in one category in the features of $\mathbf{F}'$ then the uncertainty $U_{C\text{-}of\text{-}\theta}$ will be zero.

Having defined the category uncertainty for one object over one subset of the features, we can compute the category uncertainty U_C for a categorization $\mathcal{Z}$ by averaging over all objects and over all possible subsets of $\mathbf{F}$. However, to compute such an average we must take into account the probability of having a particular feature subset $\mathbf{F}'$ available for a particular object θ_i. For a given object one set of properties may be highly salient and thus likely to be viewed, while for another object a different set of objects may be more likely available. Thus, we define the quantity $p_S(\mathbf{F}', \theta_i)$ to be the *salience probability,* where $p_S(\mathbf{F}', \theta_i) \geq 0$, $\sum_i \sum_{\mathbf{F}'} p_S(\mathbf{F}', \theta_i) = 1$. This probability is intended to reflect the likelihood of having some particular subset of features (and only that subset) available for a given object. Let $\wp(\mathbf{F})$ be the power set of $\mathbf{F}$ — the set of all subsets of $\mathbf{F}$. Then, using the salience probability as the appropriate weight for averaging the individual category uncertainties, we get the following expression for $U_C(\mathcal{Z})$:

$$U_C(\mathcal{Z}) = \frac{1}{n} \sum_{\theta_i \text{ in } \mathcal{Z}} \sum_{\mathbf{F}' \in \wp(\mathbf{F})} p_S(\mathbf{F}', \theta_i)\, H(\textit{match distribution of}\,(\theta_i, \mathbf{F}')) \quad (4.8)$$

A special case of the above equation occurs if one assumes that the salience probability is equal for all feature subsets and for all objects. In this case, since the cardinality of $\wp(\mathbf{F})$equals $2^{\|\mathbf{F}\|}$, $U_C(\mathcal{Z})$ reduces to:

$$U_C(\mathcal{Z}) = \frac{1}{n} \frac{1}{2^{\|\mathbf{F}\|}} \sum_{\theta_i \text{ in } \mathcal{Z}} \sum_{\mathbf{F}' \in \wp(\mathbf{F})} H(\textit{match distribution of}\,(\theta_i, \mathbf{F}')) \quad (4.9)$$

The above equation is used in the implementation discussed later in this chapter and in subsequent chapters. This special case was employed instead of the more general formulation because without a model of the sensing apparatus we have no basis for assigning the salience probabilities.

As a final comment about the computation of U_C, we note that the size of the power set $\wp(\mathbf{F})$ grows exponentially as the size of $\mathbf{F}$ increases. As $\|\mathbf{F}\|$ becomes only moderately large (only 15 or so), $2^{\|\mathbf{F}\|}$ becomes computationally unmanageable, since each of the subsets would be evaluated for each object. To alleviate this problem, an algorithm was implemented in which not all possible subsets of $\mathbf{F}$ are considered for each object. Rather, for each different object θ_k, a different set of subsets of features is randomly chosen for the computation. The number of feature subsets used per object can be varied, trading speed for accuracy. In the examples shown in this thesis, the sampling method was used exclusively. A comparison made between the sampling method and the exhaustive method yielded no significant differences.

We should note that the strategy of sampling the feature subsets is only valid when most features are constrained by the natural classes. Otherwise, there is a high probability that the sampled subsets will contain no useful information about the category to which an object belongs; the computation of U_C will produce erroneous results. If we require that such a sampling strategy be available to the observer, then we have placed an additional requirement on the representation: the representation must not contain too many unconstrained features. Without such a representation, the sampling strategy observer cannot recover the natural modes.

At the end of the next section, after defining the total uncertainty of a categorization, we will analyze the behavior of U_C and compare its properties to the distance metrics criticized in chapter 3.

4.4 Total Uncertainty of a Categorization

Having proposed measures for both U_P and U_C, we must now introduce the parameter λ — the relative weight of the two uncertainties — and construct a measure for the total uncertainty $U(U_P, U_C, \lambda)$. We proceed by examining how the two measures U_P and U_C behave under both ideal and pure noise conditions. By combining those results with some necessary or desirable properties that the uncertainty measure should exhibit, we restrict how $U(U_P, U_C, \lambda)$ may be constructed. We then show that a simple weighted sum satisfies these constraints.

4.4.1 Ideal categories

Our first consideration is how U_P and U_C behave in an ideal world where there are purely modal classes and features. By purely modal we mean that for each class, each feature takes on a distinct value.[10] Therefore, the features are completely predictive: knowledge of one feature is sufficient to correctly identify the class allowing the prediction of all other features. One

[10] If the current definition of a modal world appears awkward, it is because we have just confronted Watanabe's Ugly Duckling Theorem. Notice that we cannot define a purely modal world without making reference to the features. Given the discussion of natural modes in chapter 2 one would like to be able to say that some world is modal, independent of the features used to describe it. Unfortunately, as demonstrated by Watanabe, this is impossible without restricting the properties of the objects that may be used to describe the objects. For example, suppose we arbitrarily partition the world into two groups, C_a and C_b. Then, let us define a set of features $\mathbf{F}$ such that for every $f_i \in \mathbf{F}$, the objects in category C_a take on the value 1, and every object in category C_b takes the value 0. (A trivial example of such a feature is "1 if $\theta_i \in C_a$, 0 otherwise.) Then, as described by this set of features, the world would be purely modal. The only method by which we can say there exist classes in the world is by restricting the properties of consideration to be those that are of importance in the natural environment. We will return to this point later when considering how the evaluation function proposed in this section — an evaluation function derived from the goals of the observer — relates to the structure of the natural world.

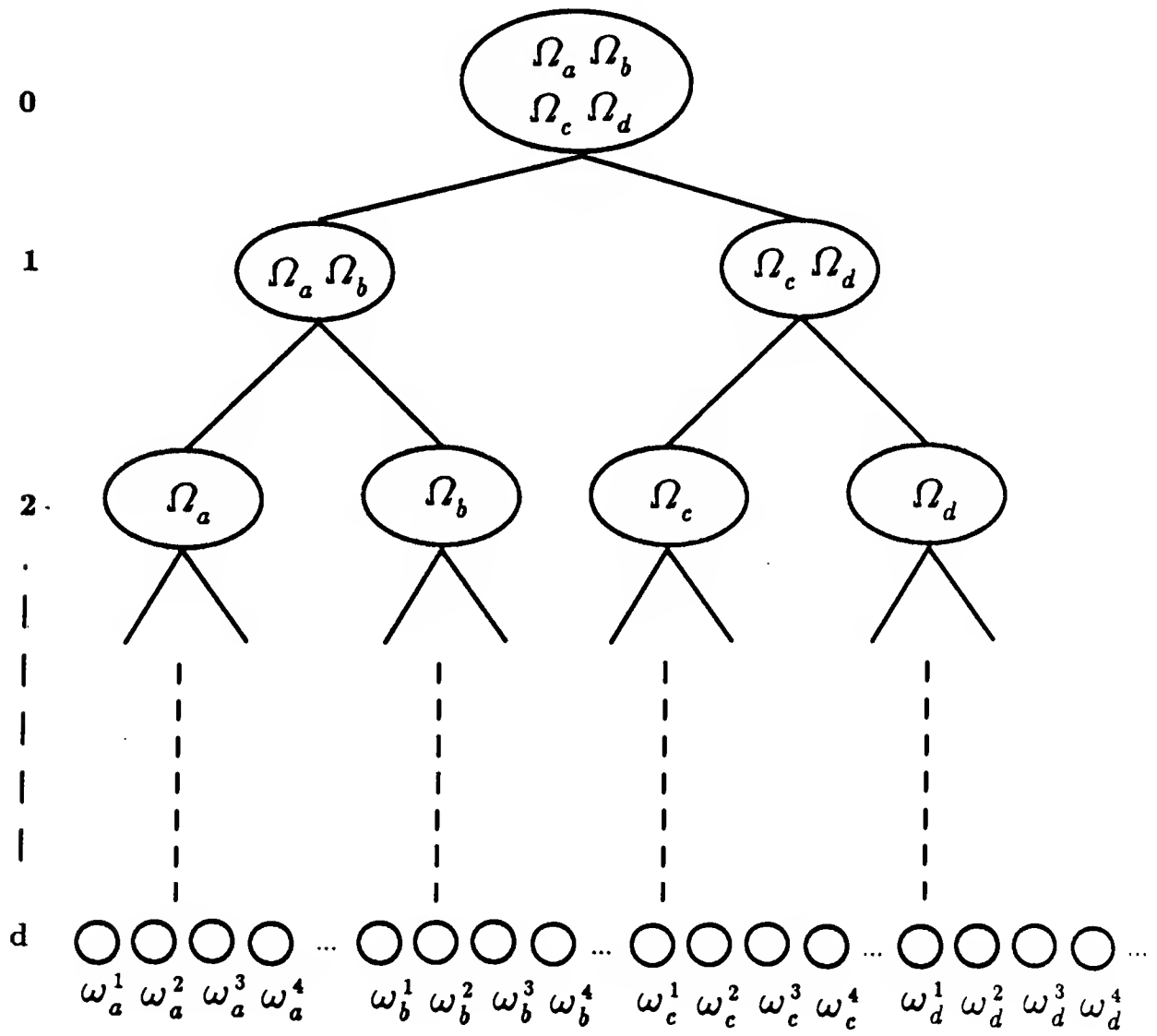
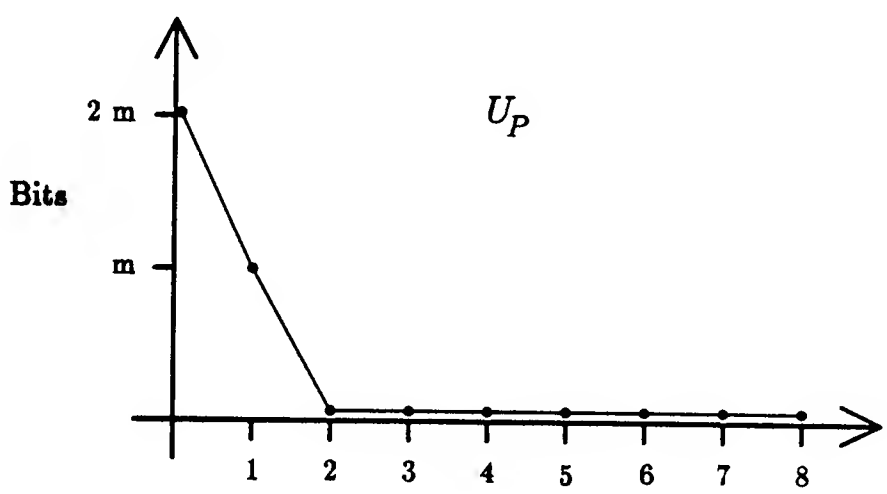


Figure 4.4: An ideal taxonomy. The hierarchy preserves the class structure exactly. At level 2, the categories of the taxonomy correspond to the classes Ω_i in the world.

possible taxonomy of such a world is shown in Figure 4.4. In this case there are four classes of objects in the world: Ω_a, Ω_b, Ω_c, Ω_d. The taxonomy is constructed such that at level 2, the categorization formed by the four categories corresponds exactly to the 4 classes in the world. At that level, all objects in a category have the same property vector, and across categories each feature takes on a different value. Because the features, f_i, used to represent the objects are purely modal there are 4 values for each feature, one value corresponding to each class.

The graphs of Figure 4.5 are the results of evaluating U_P and U_C for each of the different categorizations corresponding to a different level in the taxonomy. In these and subsequent graphs, the abscissa indicates the depth in the taxonomy. A depth of zero corresponds to the root node, where the categorization contains only one category, THING. The depth of d (where d is the deepest level of the taxonomy and $d = \log n$) corresponds to the case when each object is its own category. Notice that U_P decreases (linearly) from the root node to level 2. At the root node, all the objects are in one category, and each feature can take on one of four values; therefore $U_P = m \cdot \log 4 = 2m$, where m is the number of features. At level 1, there are only 2 possible values for each feature in each category; thus $U_P = m \cdot \log 2 = m$. Finally, at level 2, each feature is fixed to some value (in this perfectly modal situation) and there is no uncertainty about an objects properties once its category is known. Therefore, at level 2 and all subsequent levels $U_P = 0$.

The behavior of U_C may be viewed as the inverse of U_P. U_C measures the difficulty in identifying an object's category given some of its properties. In a perfectly modal world however, if no two categories contain objects belonging to the same real class, then knowledge of *any* property of an object is sufficient information to recover the category. This can be seen at levels 0, 1, and 2 in the graph of U_C. At level 0 all objects are in one category and therefore there is no uncertainty as to an objects category. At level 1, the two categories do not contain any elements of a common class: Ω_a and Ω_b are in one category; Ω_c and Ω_d, another. Thus knowledge of any property of an object is still sufficient to recover its category. Similarly for level 2, each class is in its own category and there is still no uncertainty about the category. It is only when level 3 is reached, where the classes are split among two categories, that any uncertainty arises. As the members of each class are divided among more and more categories, U_C continues to increase (linearly). At the finest categorization, at depth d, each object matches an object in



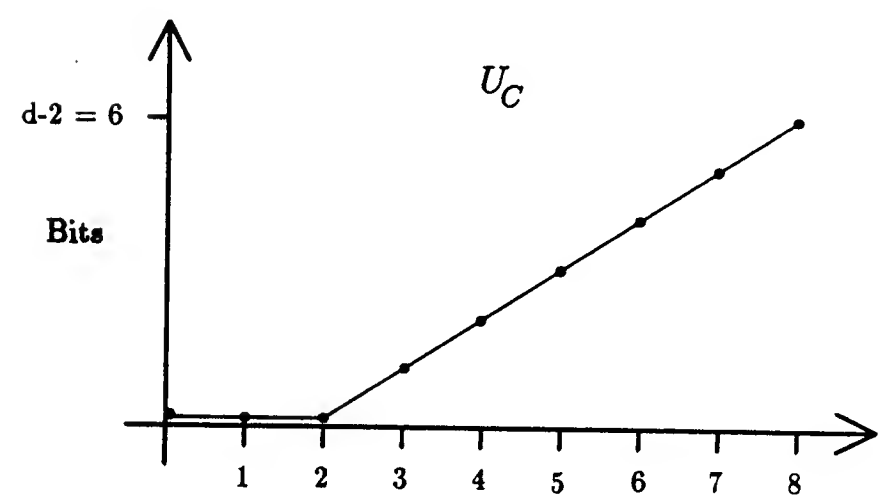


Figure 4.5: The graphs of U_P and U_C for an ideal taxonomy with four modal classes. At level 2, the categories of the taxonomy correspond exactly to the real classes in this modal world. m is the number of features. For these graphs, the maximum depth d is 8.

$n/4$ categories. Thus the maximum value for U_C is $\log(n/4) = (d-2)$.

Before proceeding to the next section, we should note that by defining the "ideal" category case we have implicitly defined natural classes to be those that are highly redundant and non-overlapping in the space of "important" properties. We will return to this point when we consider how the evaluation function derived in this section relates to the structure of the natural world. For now, we note that the discovery of categories that behave similarly to these ideal categories would permit the observer to accomplish his goals of inference. The set of categories corresponding to level two in the ideal taxonomy supports the reliable categorization of objects as well as strong inferences about the properties of an object once its category is known. The measure we construct of the total uncertainty of a categorization should be sensitive to categories of this form.

4.4.2 Random categories

We refer to a set of categories in which the features are completely independent of the categories as a *random categorization.* A simple way to consider random categorizations is to construct a random taxonomy, where the grouping of objects into a hierarchy is achieved arbitrarily (Figure 4.6). If we evaluate U_P and U_C at the different levels of this taxonomy, we would get the graphs of Figure 4.7. U_P remains constant until the number of categories becomes large and the each category no longer contains a statistical sample of the different classes of objects in the world. Similarly, U_C increases monotonically, though the rate decreases as the sampling is spread too thin. These graphs were derived experimentally through simulations.

The reason it is important to consider the random taxonomy is that such a set of categorizations represents *no structure in the data.* The categorization is useless for making any predictions. Recall that one of major criticisms of the standard cluster analysis paradigm is the inability to determine the cluster validity. Even if there are no clusters present in the data, the cluster analysis programs are obligated to "discover" categories. By requiring that our uncertainty measure have a certain pathological behavior in the case where there is no structure in the data, we will provide a mechanism by which we can determine when discovered categories are indeed valid. Note the particular form of random categorizations used here is only one (rather

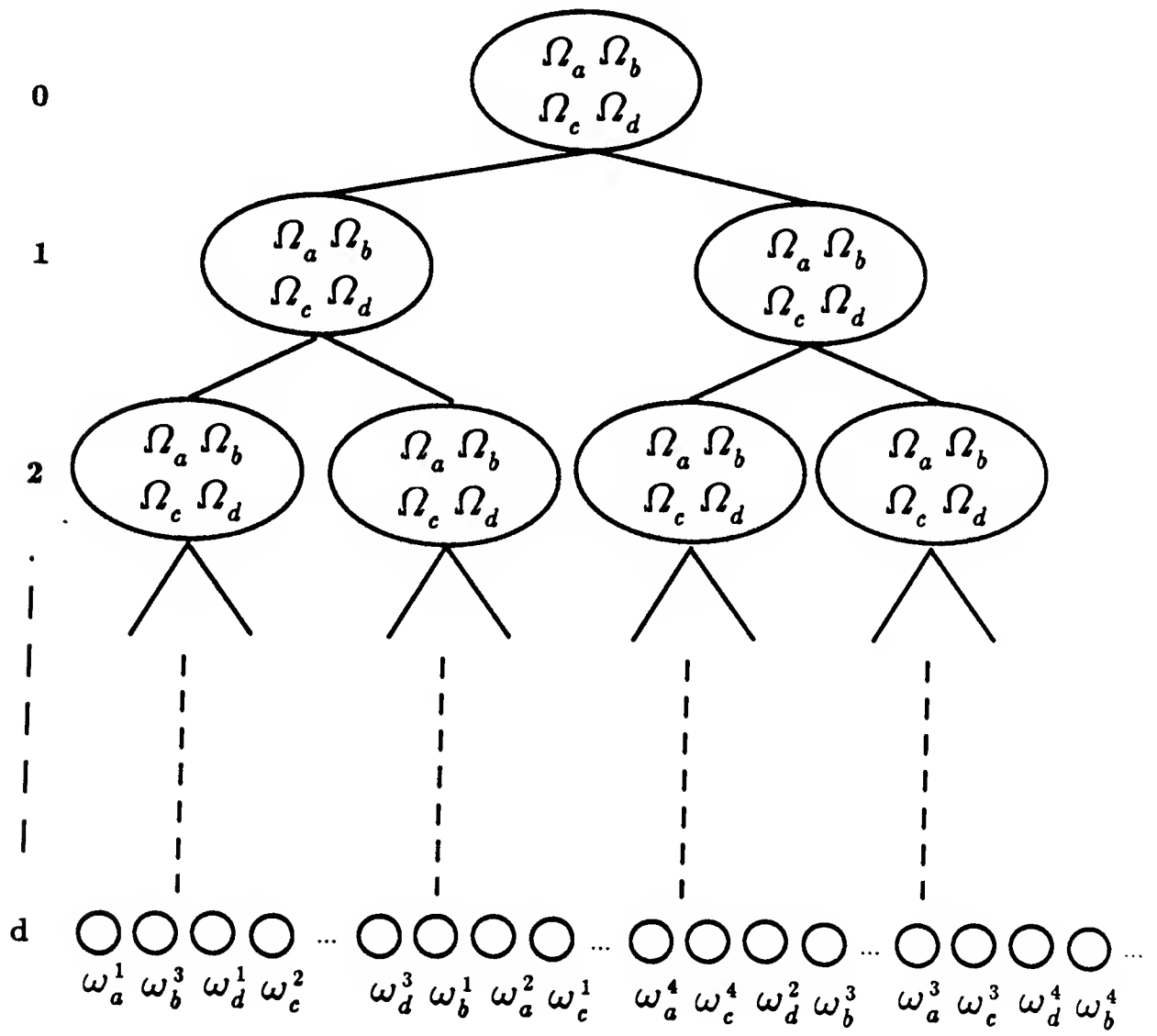
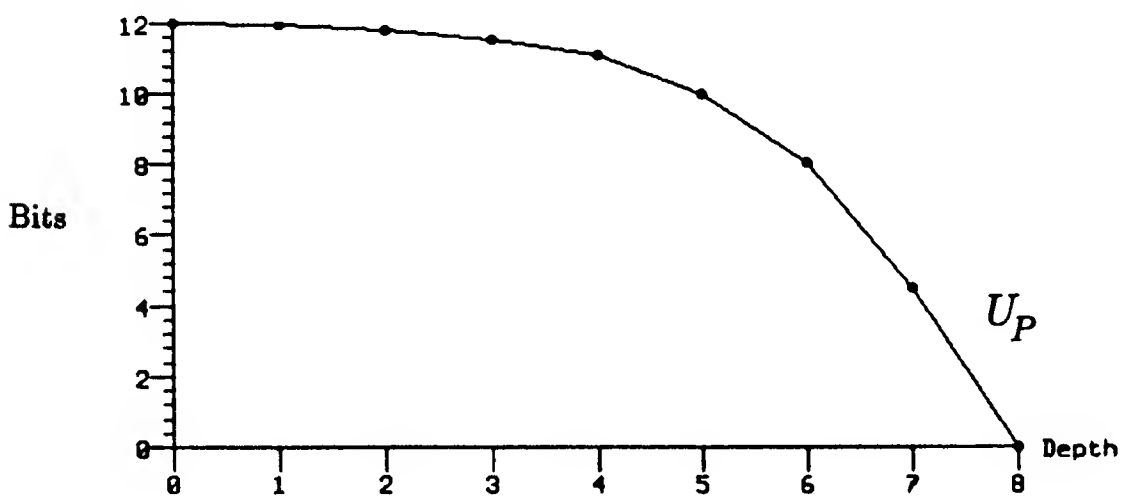


Figure 4.6: An random taxonomy. The taxonomy is created by creating a random hierarchy of objects drawn from the four classes $\Omega_a - \Omega_d$.



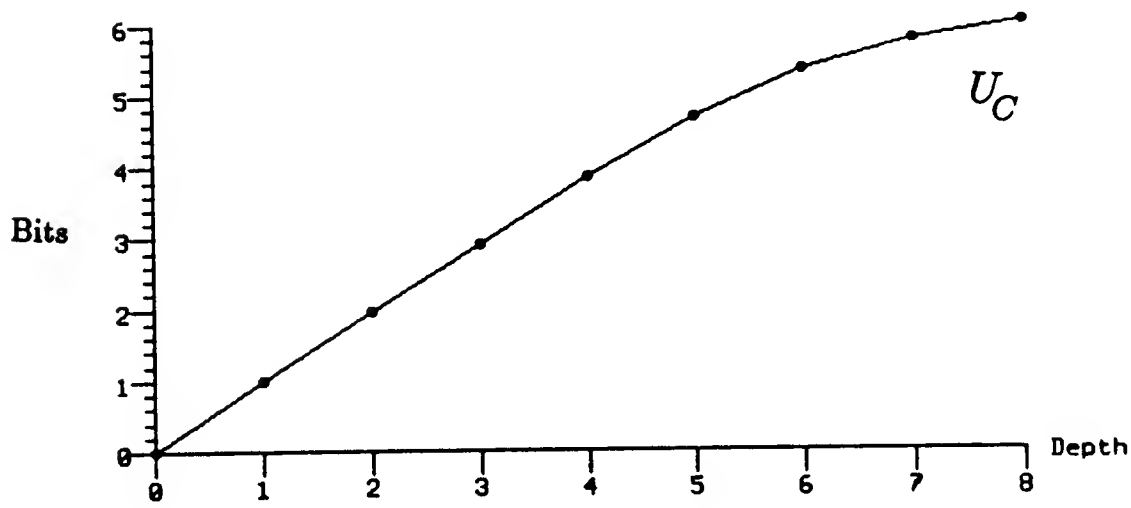


Figure 4.7: The graphs of U_P and U_C for the random taxonomy. In this example there are 6 features, and the maximum depth of the taxonomy is 8.

restrictive) model of the absence of structure. In chapter 6 we will consider an alternative form in which the observer has attempted to form the best taxonomy possible in a world which has uniformly and independently distributed features.

4.4.3 Defining $U(U_P, U_C, \lambda)$

To construct the uncertainty function U, we will first present several constraints that U must satisfy. Then, we will propose a simple measure consistent with these constraints. All of the constraints are expressed in terms of evaluating the levels of a taxonomy; the term "preferred categorization" refers to the set of categories selected when the taxonomy level that yields the lowest value of U is chosen.[11] Two of the constraints will be based upon the behavior of U_P and U_C as described in the previous section.

The first two constraints describe the behavior of U at the extreme values of λ:

1. When $\lambda = 0$, the preferred categorization should be the finest, with each object in its own category. This should be true for all possible taxonomies.

2. When $\lambda = 1$, the preferred categorization should be the coarsest, with all objects in one category. Again, this should be true for all possible taxonomies.

Another way of expressing the first constraint is that when $\lambda = 0$, the measure U should have no U_C terms, and the preferred categorization would be that which minimizes U_P. The second constraint would correspond to U being independent of U_P when $\lambda = 1$. These constraints also combine to give λ the intuitive meaning of being a relative weight between the two component uncertainties.

The next constraint expresses the desired behavior of U under the purely modal conditions:

3. In the purely modal taxonomy, the preferred categorization for $0 < \lambda < 1$ should be that which corresponds to a separate category for each class of objects in the world.

[11] If there exists more than one level of the taxonomy with the same minimum value of U, then by "preferred" level we mean that the level be one of those with the minimum value.

For example, in the taxonomy of Figure 4.4, the preferred categorization should be level 2, where each Ω_i is in its own category. This constraint states that level 2 should be preferred for *all* λ not at the extremes. If both U_P and U_C have a non-zero contribution to U, then, in an ideally modal world, the best categorization is that which selects the modal classes.

We also wish to constrain the behavior of U in the random condition of the taxonomy of Figure 4.6. Intuitively, we desire that the behavior of U in the random condition be predictable so that we can determine when we are evaluating a non-structured set of categories. We will impose this restriction in the following way:

4. In the random taxonomy (which contains no useful categories), the preferred categorization should be either the finest or the coarsest, depending on λ.

That is, for each λ, the value of U should be a minimum at one of the extreme levels of categorization. Unfortunately, this constraint can not be fully discussed until we present the concept of *lambda-space* in the next section. At that time, we will provide the intuition behind this constraint. For now we only state that a random taxonomy contains no useful intermediate structure and thus no intermediate level should be preferred.

Finally we include an constraint which allows us to compare one categorization to the next in a meaningful manner and which allows us to interpret λ as a relative weight between U_P and U_C:

5. U should be normalized with respect to the number of objects contained in the categorization.

If we were strictly adhering to the definitions provided at the beginning of this chapter we would not need to be concerned with normalization: every categorization is a partitioning of a fixed population. However, in the next chapter we will utilize the measure developed here in a dynamic, incremental categorization method. Thus we need to be able to normalize for the number objects contained in a categorization. Also, to interpret λ as the relative weight between U_P and U_C we must make their scales commensurate.

Combined, these constraints restrict the functional form of U; we shall propose a simple measure for U which satisfies these five constraints. Afterwards, we will compare this measure with some of the category metrics discussed in chapter 3.

We first need to introduce a normalization coefficient which will make the measure independent of the number of objects in a categorization. Note that given "enough" objects per category, U_P is independent of the number of objects in a categorization, since it depends only on the entropy of the properties. U_C, however, may depend critically on the number of objects: given more more objects we can create more categories and make the number of possible category matches of an object be arbitrarily large. Therefore we need to scale U_C appropriately for the number of objects. Also, though by design both U_P and U_C are unitless (or sometimes said to be in units of information referred to as bits), they are not of the same range. The maximum value for U_P is unrelated to the maximum value for U_C. Therefore to make them commensurate we will scale the normalized U_C by the maximum U_P.

We compute the normalization coefficient as follows: Suppose we are given some categorization $\mathcal{Z}$ to evaluate. Let us construct two new categorizations from $\mathcal{Z}$. Define $Coarsest(\mathcal{Z})$ to be the categorization formed by placing all the objects of $\mathcal{Z}$ in one category. Analogously, define $Finest(\mathcal{Z})$ to be the categorization formed by placing all the objects in $\mathcal{Z}$ into separate categories. We define a normalization factor $\eta(\mathcal{Z})$:

$$\eta(\mathcal{Z}) = \frac{U_P(\, Coarsest(\mathcal{Z})\,)}{U_C(\, Finest(\mathcal{Z})\,)} \tag{4.10}$$

By dividing by $U_C(\, Finest(\mathcal{Z})\,)$ we compensate for the number of objects; the numerator makes the scale the same as that of U_P.

Finally, we can propose our measure for the total uncertainty of a categorization:

$$U(\mathcal{Z}) = (1-\lambda)\, U_P(\mathcal{Z}) + \lambda\, \eta(\mathcal{Z})\, U_C(\mathcal{Z}) \tag{4.11}$$

The total uncertainty of a categorization is simply the weighted sum of U_P and U_C, where U_C has been scaled to be of the same range as U_P; the parameter λ controls the relative weights.

It is easily shown that equation 4.11 satisfies the behavioral constraints 1–4. Constraint 1 holds because when $\lambda = 0$, $U(\mathcal{Z}) = U_P$, and U_P is at a minimum (in fact zero) at the finest categorization where each object is in its own category. Constraint 2 follows analogously. Constraint 3 is also satisfied: if $0 < \lambda < 1$, then U is at a minimum (zero) when both U_P and

U_C are zero. As shown in Figure 4.5, this occurs only at the desired natural class categorization. In fact, since zero is the absolute minimum for U, in an ideally modal world, the categorization which corresponds to the natural classes is the best *possible* categorization, not simply the best level in some taxonomy.

Constraint 4 holds because of the concavity of both U_P and U_C in the random condition. Because the linear sum of two concave functions is also concave (Luenburger, 1986), U is guaranteed concave for the random case. Therefore, for all λ, the minimum of U is at one of the extreme levels of the taxonomy.

Of course, one could construct a more complicated measure; given that the proposed measure satisfies the imposed constraints and that it is similar to standard functionals for combining constraints, there is no apparent reason to do so. In the next section we will compare the properties of $U(\mathcal{Z})$ with some of the distance metrics discussed in chapter 3.

4.4.4 Uncertainty as a metric

Having provided a formal definition for the uncertainty of a categorization, we can now compare this function to the distance metrics discussed in chapter 3. Specifically, we should address the criticisms raised concerning the use of distance metrics to define object categories.

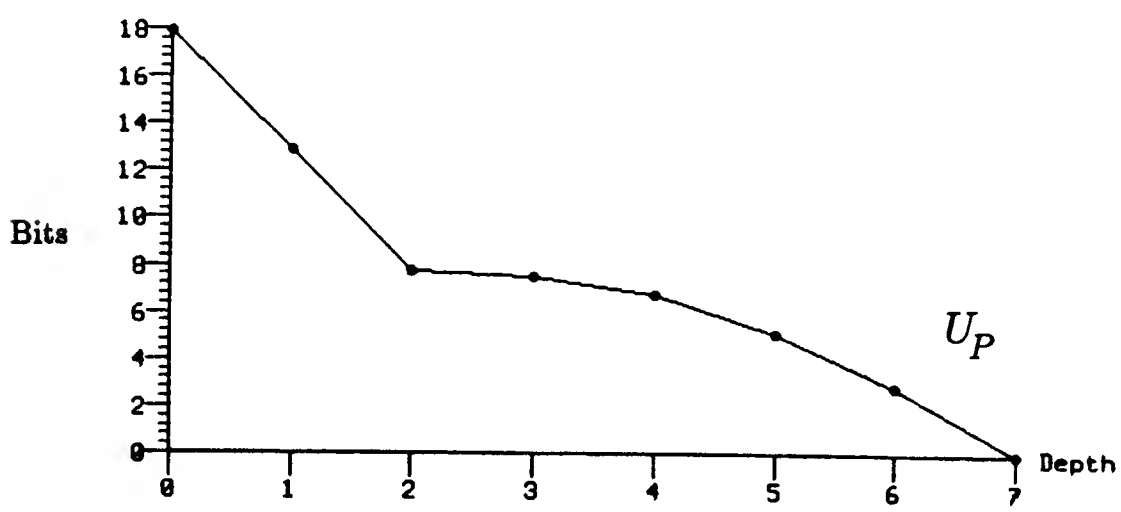
First, notice that although we do not explicitly define the distance between two objects, the property uncertainty functions do provide an implicit measure for comparing two objects. In particular, the entropy function imposes a Hamming-like distance between objects since the entropy measures are sensitive to the exact matches between feature values. As mentioned in chapter 3 such measures are sensitive to the resolution of the features. For example, if a feature is continuously valued (e.g. "length") and is histogrammed into fine-grained buckets, then all objects will take on a different value. In this case the feature will be able to convey no information about classes in the data. Thus, using these entropy measures requires that some of the features of the representation be suitably chosen to convey the distinctions between different classes of objects.

For two reasons, the above restriction does not significantly reduce the utility of the categorization evaluation function. First, it is necessarily true that the observer must encode relevant sensory information if he is to dis-

cover natural classes of objects. If the only properties of objects used for categorization were those completely independent of the "type" of object, then no interesting properties could be predicted from those observations. If we invoke the power of evolution in the design of the observer, then we expect that the observer would be provided with a set of features sufficient to determine the true class of the object and thereby granting him the ability to form an appropriate set of categories. In chapter 6 we will demonstrate how the observer could use an initial set of useful features to evaluate the predictive power of a new feature. But, a sufficient initial set must be provided.

Second, the nature of the entropy measures is that not all of the features need be constrained. Unlike standard distance metrics where long inter-object distances in a few dimensions can mask clustering in other dimensions, entropy measures are statistical in nature and and can detect structure in separate dimensions. U_P was designed to treat the features independently and U_C considers object matches along different subsets of features. We can demonstrate this behavior by reconsidering the ideal taxonomy of Figure 4.4. Let us assume we have the same taxonomy of objects and the same modal features. However, this time we shall include several noise features — features whose values are independent of the object class. Figure 4.8 displays the results of evaluating U_P and U_C for the different levels of a four class taxonomy. Notice that both uncertainties are no longer zero at the modal level; the increase in uncertainty is caused by the noise features. However, there is still a significant change in the behavior of both U_P and U_C at level two. As the the number of noise features is increased the change in the slope of the curves diminishes; when there are many more noise features than modal features the graphs approach those the random taxonomy in Figure 4.7.

The ability to still detect structure in the presence of unconstrained features also allows the entropy measure to be used when different features are constrained for different classes of objects. Therefore, there is no requirement of using only features constrained in all classes. Metrics based on the within-class and between-class scatter matrices are unreliable in the presence



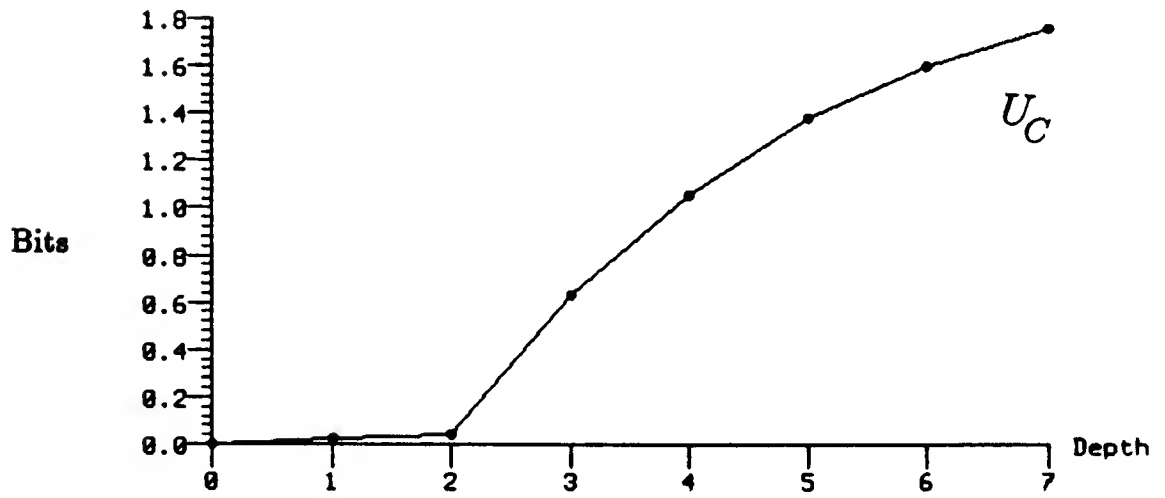


Figure 4.8: The graphs of U_P and U_C for an ideal taxonomy of four classes, but with noise features added. At level 2, the categories of the taxonomy correspond exactly to the real classes in this modal world. With the addition of noise, the curves are no longer the simple linear functions of the pure modal case, but there is still a definite break at the natural level.

of unconstrained dimensions. Furthermore, the need to modify the distance metric as one moves from one region of feature space to another becomes less pressing since the measure can respond to one set of properties for one class, and another for a different class. Thus using entropy functions eliminates several of the difficulties associated with using distance metrics for categorization.

We should note that the total uncertainty evaluation function derived in this section is analogous to the modified measure of collocation discussed in chapter 3. Had we combined the measures U_P and U_C via an exponentiated product $(U_P^{(1-\lambda)} U_C^{\lambda})$ we would have produced a measure directly related to the extended collocation function:[12]

$$K'_{C_j,f_i} = P(C_j|f_i)^{\lambda} \cdot P(f_i|C_j)^{(1-\lambda)}$$

Thus, we have incorporated the lesson of basic level categories in our measure of uncertainty: a measure designed to evaluate basic categories must consider both the cue validity of the features and the internal similarity of categories.

Finally, we note that the category evaluation function derived in the previous section explicitly measures how well the observer can accomplish his goals of inference. Recall that one of the criticisms of the cluster analysis paradigm was that it made no sense to consider the utility of the recovered classes. By definition the classes recovered were those which minimized the particular evaluation function. Whether these categories were appropriate for some task depended on how well the requirements of the task mapped onto the clustering criteria used. In our case, we have constructed a criteria that *directly measures the utility of a categorization for the task of making inferences about objects.* If one believes that the goal of object recognition is to make inferences about objects, then the set of categories selected by the categorization criteria $U(\mathcal{Z})$ is appropriate for recognition.

4.5 Natural Categories

The evaluation of a categorization proposed in the previous section is based upon the goals of the observer; a categorization which has a "low" measure

[12]Since both U_P and U_C can be zero, this particular expression would be ill-defined unless other parameters or constants are added.

of uncertainty U should permit the observer to perform his necessary inference tasks successfully. Furthermore we have constructed the measure such that in an suitably defined ideal world, the measure prefers a modal categorization over any other. However, we have yet to mention the world of objects which the observer is going to categorize. In section 2 we introduced the Principle of Natural Modes as the basis for the categorization process. The claim was made that the reason it is plausible that the observer could predict unobserved properties from observed properties was that there were constraints acting in the world which caused redundancies between observed and unobserved properties to be present. How do we incorporate the idea of natural classes into our evaluation of categorizations?

4.5.1 Natural classes and natural properties

The first question to be considered is how does the proposed evaluation function behave if the categorization being measured does reflect the natural modes present in the world. Recall that U was constructed such that in an ideally modal world, the categorization corresponding to the natural classes would be the preferred. In the modal world, each feature was completely diagnostic. Whether the proposed evaluation function will be able to capture the structure present in the natural world will depend upon the diagnosticity of the chosen representation. That is, the representation must be chosen such that the constraints imposed by the natural object processes are reflected in the properties measured.

An example will help to illustrate this point. Consider the case where we have some class of objects where the aspect ratio (ratio between the length and the width) is fixed by the process which generates that class. Suppose that both "length" and "width" are features measured about the object, but that aspect ratio was not. Our measure of total uncertainty would not be sensitive to this constraint being present in the class. If, however, aspect ratio was a feature, then this constraint would be reflected in the measure of both property and category uncertainty; categorizing that particular class of objects separately would reduce the uncertainty measure. Thus we rely on the choice of features (which define those properties which are observed) being appropriate for measuring the constraint that is found within classes.

Note, that we although require that the representation be sufficiently restricted so that the differences between classes are made explicit, we do

not require that irrelevant information be prohibited from the representation. Recall the graphs of figure Figure 4.8. These results demonstrate that the uncertainty measure can still detect the ideal modal structure even when a significant portion of the property description is generated randomly. When we test the evaluation function on real data in the next section, we will discover that real property descriptions behave in a manner similar to those produced with modal and noise features.

4.5.2 Lambda stability of natural categories

One might consider a sufficiently low measure of total uncertainty U to be simple indicator of a correct natural mode categorization. Unfortunately, this requires having some absolute metric of uncertainty for categorizations. For example, consider again the taxonomy of Figure 4.1, and recall that we are considering the categorizations which correspond to the different levels in the tree. Let us assume that for some $\lambda = \lambda_0$ the third level yielded the least total uncertainty U. How can we know whether this level reflects modal structure in the objects of the taxonomy, or if this is simply some arbitrary categorization which just happens to evaluate to the lowest uncertainty for the given λ? This question is analogous to the question of cluster validity raised in chapter 3.

Let us assume that we have been given a taxonomy such as Figure 4.1 and that for some discretized range of λ, $0 \leq \lambda \leq 1$, we have selected the categorization corresponding to the level in the taxonomy which minimizes the total uncertainty. We can plot the results of this procedure in a *lambda space* diagram as illustrated in Figure 4.9. Notice that for $\lambda = 0$ the best categorization is that which places all the the objects in their own category. Likewise, $\lambda = 1$ selects the top level, where the only category is THING. The question we must consider is how does the selected level change as λ varies? By design, we know that in the ideally modal case the categorization corresponding to the modal classes will be preferred for all λ, $0 < \lambda < 1$. But what about "real" natural classes?

In Figure 4.9 the hypothetical behavior is that over some (wide) range of λ, the preferred categorization remains the same. We refer to this behavior as λ-*stability*. The occurrence of λ-stability indicates that the categories selected for that range of λ are robust with respect to deviations in the relative weight between property uncertainty and category uncertainty. Therefore,

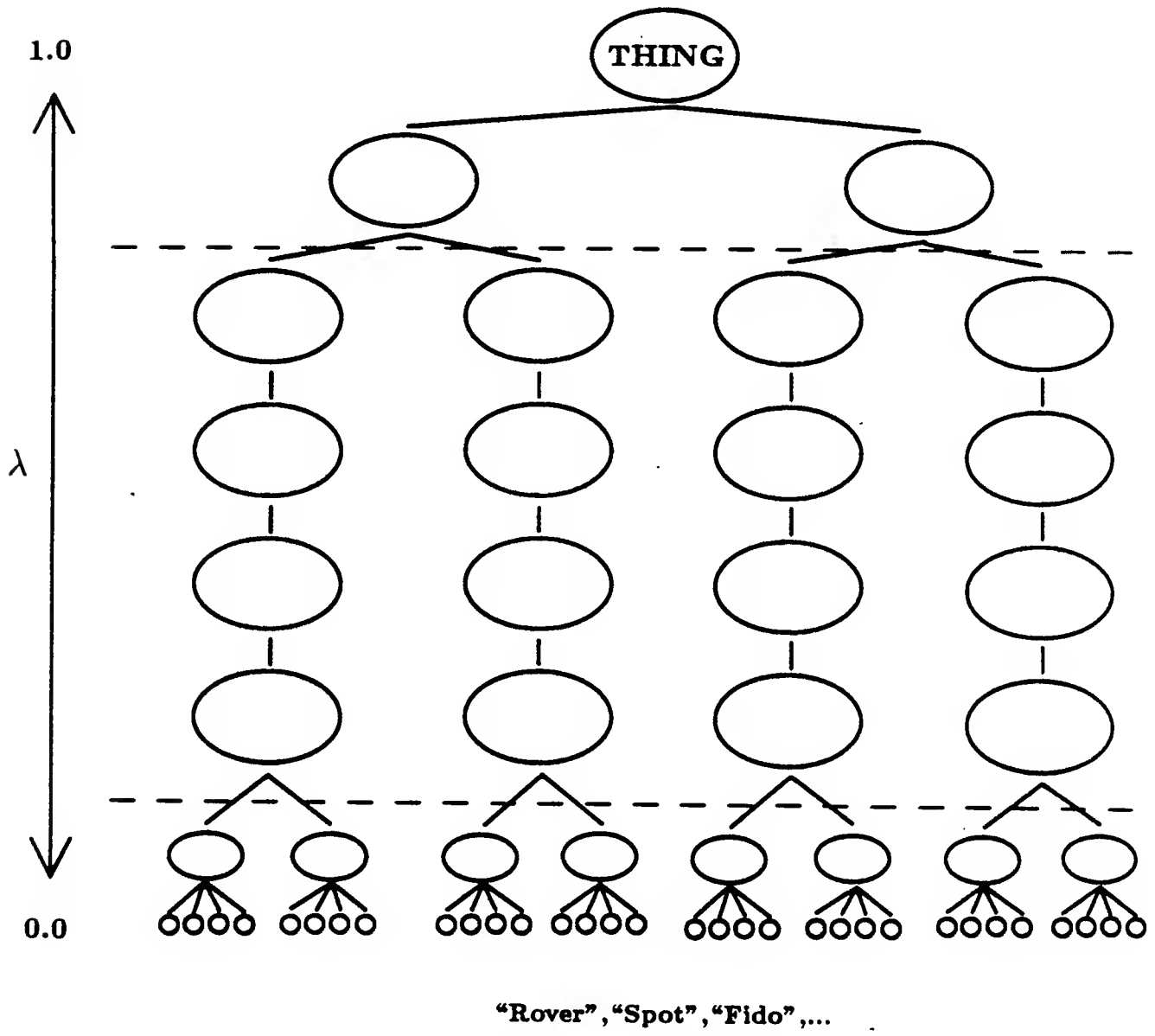


Figure 4.9: Stable λ-space diagram. For each λ in some discrete range between 0 and 1, the level of the taxonomy is selected which minimizes the total uncertainty.

they represent an actual structuring of the objects, not an arbitrary minimization of the uncertainty function. If the categorization was just the arbitrary minimum, we would expect that the preferred level would change as variations in λ drive the minimum solution toward either of the two extremes. In chapter 6 we will consider the question of λ-stability in greater detail and will consider how one can use the measure of stability as a tool to accomplish other important tasks related to the categorization process, e.g. evaluating the utility of a new feature.

Another possible behavior as λ varies is illustrated in Figure 4.10. In this case, the only stable points are the two extreme categorizations. Recall that the fifth constraint on the construction of the total uncertainty function U was that if there was no internal internal structure of a taxonomy, then the categorization which should be preferred should be one of the two extremes. The intuition behind this constraint is the following: Consider a taxonomy which is created randomly. Therefore in terms of the uncertainties measured, each level of aggregation represents the same trade-off between category homogeneity and category overlap, between property uncertainty and category uncertainty. Now let us describe λ as a pressure to move up the taxonomy. The larger λ gets, the easier it is to trade the gain in property uncertainty for the reduction in category uncertainty which occurs when categories are merged. When λ starts at 0, the preferred categorization is the finest partition, with each object in its own category. As we initially increase λ the preferred level of categorization does not change because there is insufficient pressure to overcome the increase in property uncertainty which occurs by randomly combining objects. Eventually, however, λ is great enough that the first level of merging takes place. But, as stated, in a random taxonomy each level of merging is the same amount of trade-off between property uncertainty and category uncertainty. Therefore once λ is great enough to prefer level $d-1$ over level d, it is great enough to prefer level $d-2$ over $d-1$, continuing until level 0 is reached. Therefore, at some critical λ the preferred level of categorization moves immediately from the lowest level to the highest level.

As mentioned earlier, the noise taxonomy is only one possible null hypothesis about the absence of structure in a set of categories. In chapter 6 we will again return to the question of category validity, and compare the results of ideal (purely modal) worlds, real worlds, and a different case of a

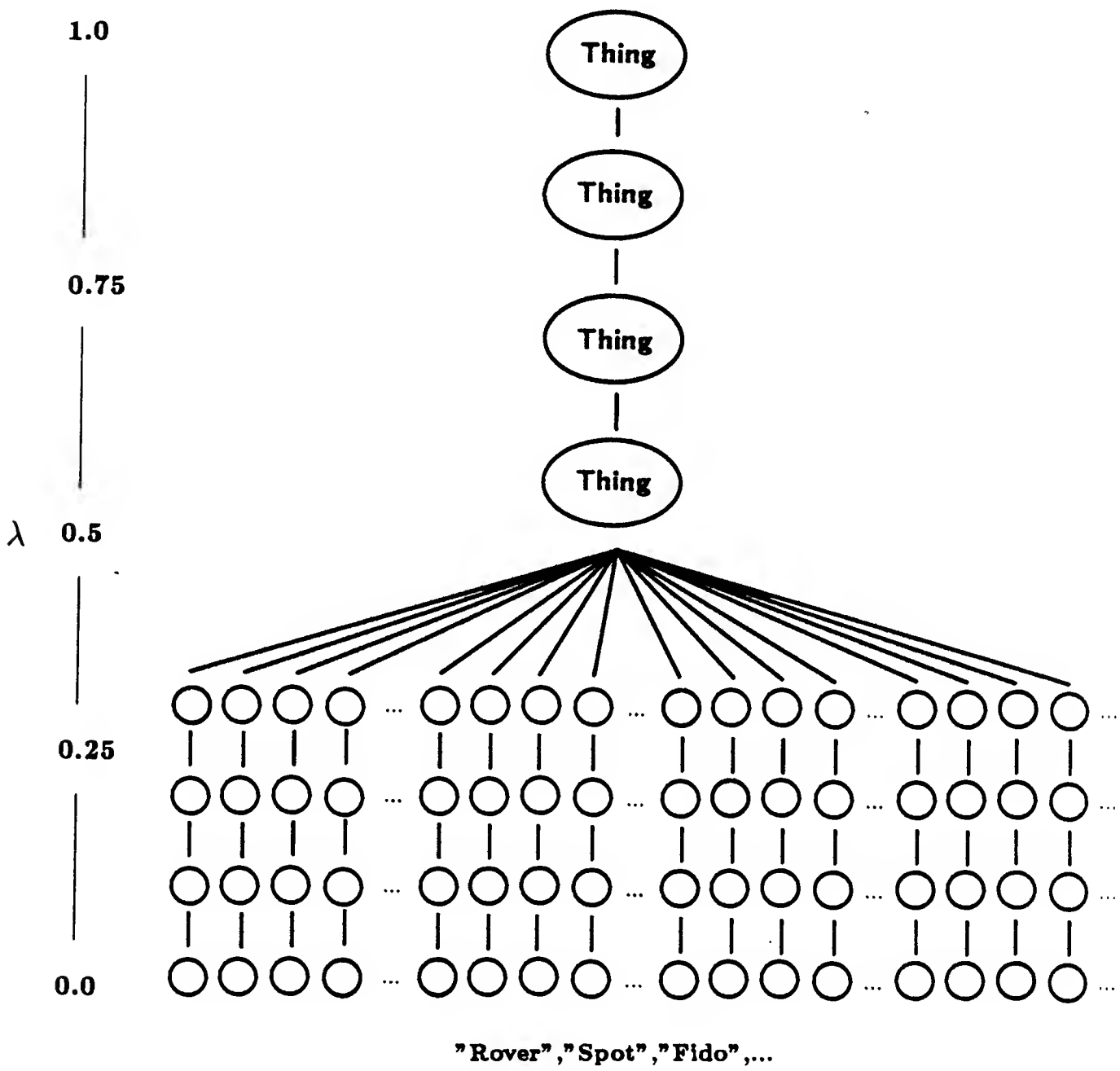


Figure 4.10: Degenerate λ-space diagram. The only preferred levels of categorization are the two extremes indicating no coherent internal structure of the taxonomy.

noise world. At this point we have a certain degree of confidence that the total uncertainty evaluation function will be useful for measuring how well a categorization reflects the natural mode classes. Let us now test the function on some real data taken from the domain of leaves.

4.6 Testing the measure

To test the behavior of the uncertainty measure U of equation 4.11 we need a sample domain of objects which satisfies the criteria of having well defined natural classes. Of course, this criterion implies a previously agreed upon method of categorization which produces natural categories. Therefore we must make the a priori assumption that the science which studies the domain establishes a baseline to which we compare our evaluation. The validity of this assumption depends upon how well the science understands the processes which determine the structure of the objects in the domain.

4.6.1 Properties of leaves

The sample domain used is that of leaves. The categorization of trees according to their leaves is a well developed discipline, and there exist agreed upon categories. The source of the leaf data is Preston's North American Trees [Preston, 1976].

In order to apply the uncertainty measure to our domain, we must create a property based description of the leaves. But which properties should be used? Are arbitrary features permissible, or should our choices be somehow restricted? To proceed we must delineate some criteria by which to choose our feature set.

The first restriction we will impose is (well-defined) computability. By this restriction we mean that if some property is going to be included in the representation for leaves, then one must be able to provide a plausible method for computing this property directly from the the physical structure of the leaf. The reason that this restriction is important is that otherwise the property "oak-ness" — how much the leaf looks like an oak leaf — would be an acceptable property. If such features are permitted, then categorization reduces to providing such features which are characteristic functions for each category. As Fodor has commented: "if *being-my-grandmother* is a legiti-

mate feature then it's pretty clear how to do object recognition." [Personal communication.] In our leaf example we will further restrict our properties by requiring them to be computable from information recoverable from an image of the leaf, precluding features such as stickiness or scent. Our reason for doing so is simply that *visual* categorization is of primary interest.

We note here that in the following examples, we did not actually provide the system with a sensory input (e.g. images). Rather, after deciding which features were to be used, property vectors were given directly. The motivation for eliminating the property computation step is that we are not interested in how well we can provide algorithms capable of measuring the properties. Our interest lies in seeing how well these properties can be used to measure structure in categories.

Having restricted our properties to being well-defined computations, we still have to choose which of these properties should be used. For our first examples we will use the features normally used by tree entymologists to classify leaves. By using this set we are guaranteed to have a set of features which contain sufficient information to distinguish between the classes of leaves. Of course, these features tend to be highly diagnostic as they are used by botanists for the express purpose of classification; however, some of the features overlap the species considerably (e.g. "length") . Also, we will consider the case of adding some noise features to the descriptions: features whose values are independent of the type of leaf. Those results will demonstrate a graceful degradation in the ability of the uncertainty measure to detect the correct categories.

Table 4.1 is a list of features used to describe the leaves, the values in the range of each feature, and a brief description of how they would be computed from an image. One of the features normally used by botanists to describe leaf categories is "shape," where several distinct shapes types are used as primitives. Since this feature bordered on not being a well defined computation, it was replaced with the three features of width, length, and flare, where flare is the direction and degree of tapering of the leaf.

Using these features, we can describe several *leaf-specifications*. A specification is a set of values for each feature which would be consistent with the description of a leaf species found in Preston [1976]. Table 4.2 provides the set of specifications used for the examples used here. Several points should be made about the features. First, there are features which are not highly

Feature	Values	Method of Computation
Length	{1,2,3,...,8,9}	Measure directly
Width	{1,2,3,...,6,7}	Measure directly
Flare	{-2,-1,0,1,2}	Fit best ovoid
Lobes	{1,2,3,...,8,9}	Filter and count
Margin	{Entire, Crenate, Serrate Doubly Serrate}	Fractal dimension of edge
Apex	{Rounded, Acute, Accuminate}	Curvature of tip
Base	{Rounded, Cumeate, Truncate}	Curvature of base
Color	{Light, Dark, Yellow}	Measure green component of color

Table 4.1: Leaf features and values.

constrained by the specifications and which have high inter-category overlap, e.g. length and width. Second, the specifications are disjoint; no leaf could be constructed which satisfies more than one specification. Finally, there is no small subset of features (less than 4) which would be sufficient to distinguish between the species.

A "leaf generator" has been constructed which takes as input a leaf specification and produces a property vector consistent with the specification. These property vectors are the "objects" used for all of the experiments.

4.6.2 Evaluation of taxonomies

To begin our testing of the behavior of the uncertainty measure U, let us consider the task of evaluating levels of a taxonomy. Although this presupposes being provided a taxonomy to evaluate, we will be able to check that the measure U behaves as predicted. We will be able to compare the situation in which there is structure in the taxonomy to that when a taxonomy is created randomly. Later, we will address the problem of discovering natural categories in an object population.

Figure 4.11 is an example taxonomy. At the bottom of the taxonomy are the leaves; in this case there are the four types of Oak, Maple, Poplar and Birch. These single object categories are combined to form the next level of categories, continuing on until all the leaves are in one category. The letters written in each node indicate the types of leaves contained in

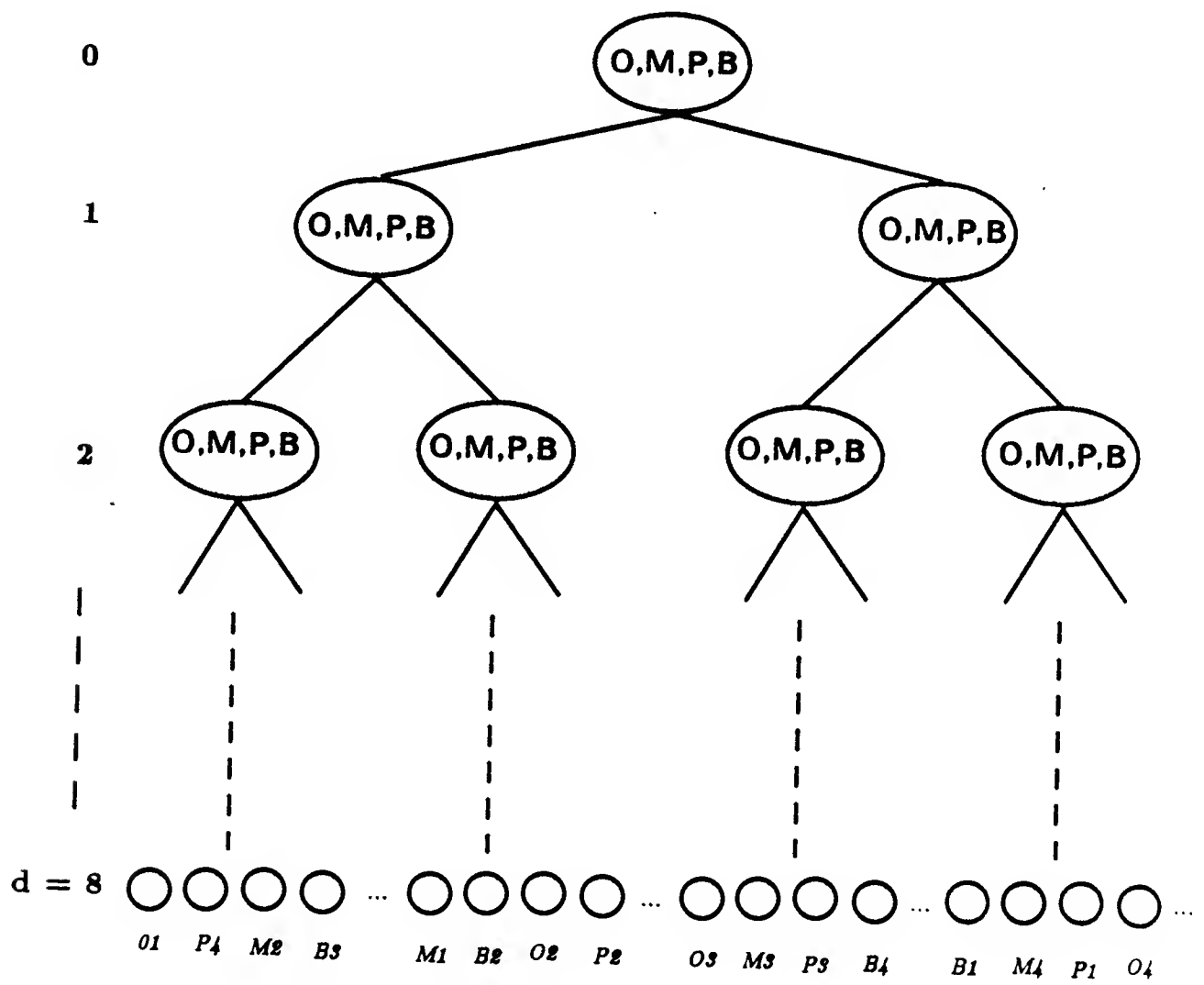


Figure 4.11: A jumbled taxonomy of Oak, Maple, Poplar, and Birch leaves. Leaves are randomly combined to form higher category.

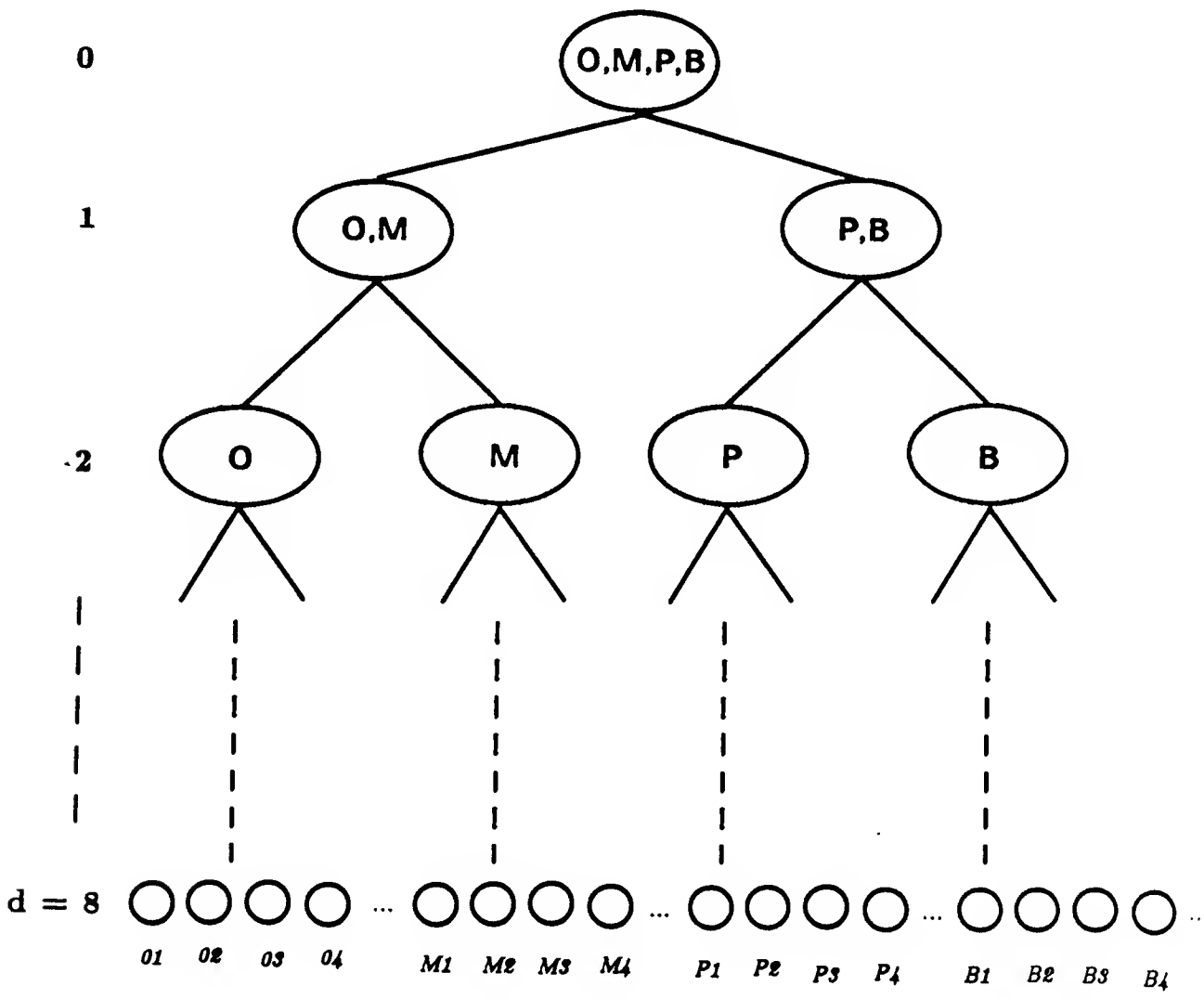


Figure 4.12: An ordered taxonomy of Oak, Maple, Poplar, and Birch leaves.

	Length	Width	Flare	Lobes	Margin	Apex	Base	Color
Maple	{3,4,5,6}	{3,4,5}	0	5	Entire	Acute	Truncate	Light
Poplar	{1,2,3}	{1,2}	{0,1}	1	Crenate, Serrate	Acute	Rounded	Yellow
Oak	{5,6,7,8,9}	{2,3,4,5}	0	7,9	Entire	Rounded	Cumeate	Light
Birch	{2,3,4,5}	{1,2,3}	0	19	Doubly-Serrate	Acute	Rounded	Dark
Cottonwood	{3,4,5,6}	{2,3,4,5}	2	1	Crenate	Acuminate	Truncate	{Light,Dark, Yellow}
Elm	{4,5,6}	{2,3}	{0,-1}	1	Doubly Serrate	Accuminate	Rounded	Dark

Table 4.2: Leaf specifications for several species of leaves. A leaf generator was designed which created property vectors consistent with the different specifications.

that node. The bottom nodes (the leaves of the tree, if you will) represent single instances of leaves. Figure 4.11 is a random taxonomy, where the leaves were arbitrarily combined to form higher categories. There is a total of 9 levels (0–8) indicating 256 leaves, 64 of each type. For comparison, Figure 4.12 is an ordered taxonomy of the same leaves in which the nodes have been constructed so as to preserve the natural classes of the species. The first question we will consider is how the two components U_P and U_C of the total uncertainty measure behave as we evaluate different levels of these two taxonomies.

4.6.3 Components of uncertainty

In the graph of Figure 4.13 the quantities of U_P and normalized U_C are plotted as a function of depth in the taxonomy. A depth of zero corresponds to the top level of the taxonomy with only one category; a depth of 8 (because there were 256 leaves in this example) is the finest categorization. Both curves are monotonic in depth as predicted when the quantities were derived. Notice that both curves vary smoothly, indicating no special level in the taxonomy. Because the taxonomy was created by randomly combining leaves, no level contains any more structure than any other level.

Now let us consider the taxonomy in Figure 4.12. In this case the taxonomy segregates the different types of leaves at level 2, with the finer divisions

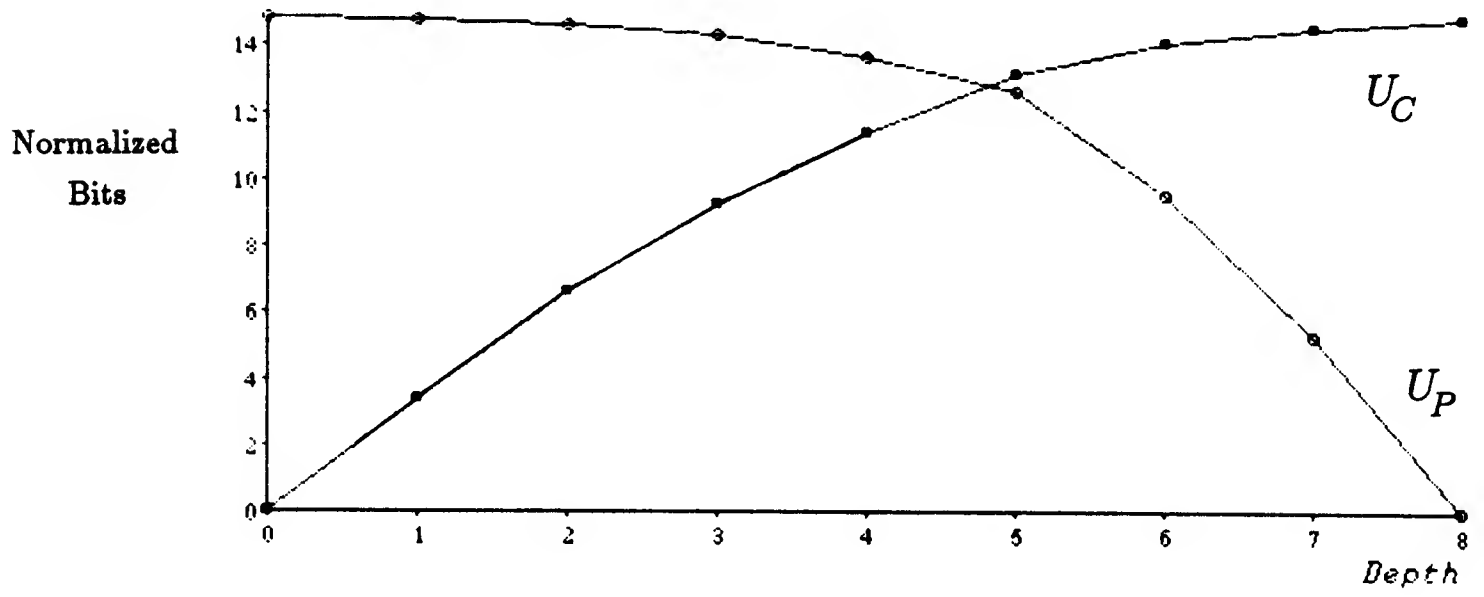


Figure 4.13: The evaluation curves for the jumbled taxonomy. Plotted are U_P and the normalized U_C as a function depth. The normalization factor causes the scales of the two graphs to be the same. Both curves change smoothly, indicating no special level within the taxonomy.

below that level being made randomly. The evaluation curves for this taxonomy are plotted in Figure 4.14. Now the curves no longer vary smoothly, but have a distinct break at the second level where the different types of leaves are segregated into different categories. Let us trace each of the curves. As predicted, the property uncertainty starts at a maximum at level 0. Splitting into two categories, each containing two types of leaves, significantly reduces the property uncertainty since knowing which of the two categories a leaf comes from restricts its properties to being of one of two types of leaves instead of four. The next split into four categories (at level 2) causes a similar decrease in property uncertainty. However, after level two, there is no significant decrease in property uncertainty because a category which has 32 leaves of one type has not much less property uncertainty than a category which has 64 leaves of that type. The property uncertainty remains almost constant until end effects occur and there are few leaves per category.

The category uncertainty U_C also markedly changes its behavior at the second level. As expected, at level 0, where all objects are in one category,

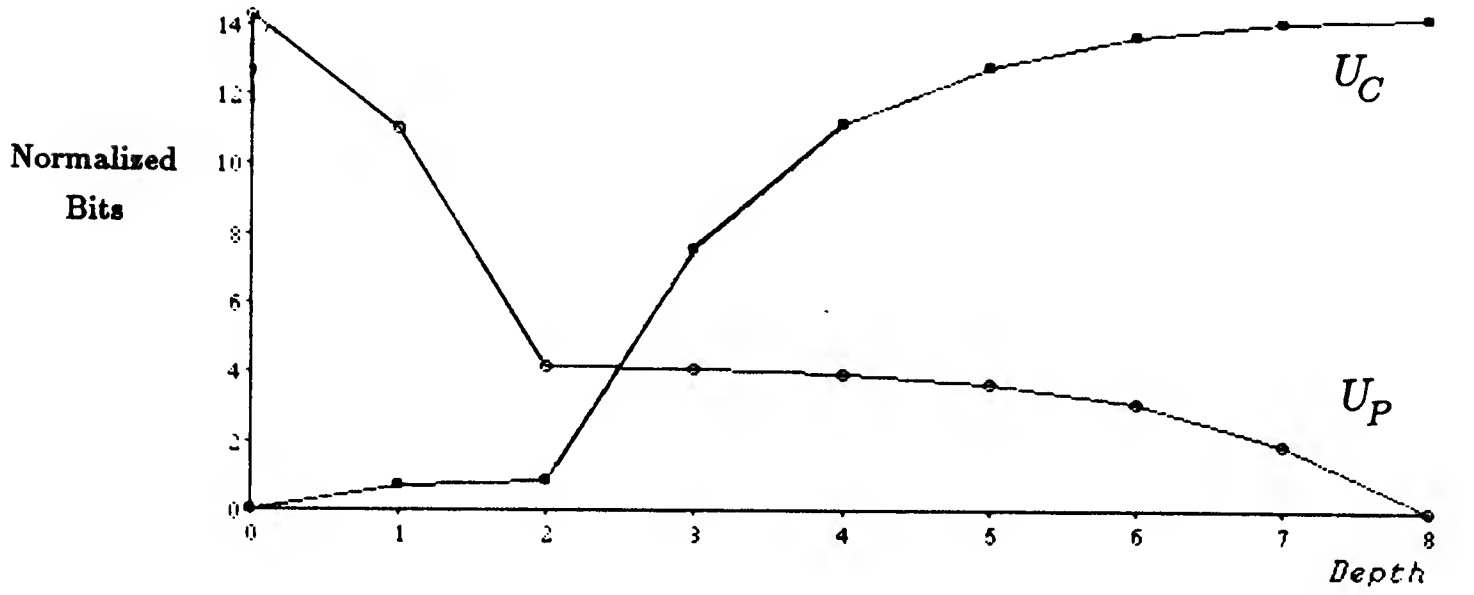


Figure 4.14: The evaluation curves for the ordered taxonomy. Plotted are U_P and the normalized U_C as a function depth. For the structured case of the ordered taxonomy the level at which the species of leaves are separated — level 2 — shows a marked break in both U_P and U_C.

there is no category uncertainty. Splitting the leaves into two categories which do not share leaves of the same type produces only a marginal increase because the categories are quite distinct and partially described leaves are still easily categorized. Splitting into the four leaf types similarly adds little category uncertainty. However the next split causes an abrupt increase in category uncertainty. This is caused by the fact that now there are two categories containing leaves of each type. Therefore a partially described leaf will often match leaves in more than one category, yielding a high value in category uncertainty. As the categorization gets finer U_C continues to increase.

It is important to notice that the graphs of Figure 4.14 are similar to those of Figure 4.8. In that example we evaluated a taxonomy of purely modal classes and features, but with the addition of several noise features. This similarity indicates that features which are not purely modal — they do not perfectly discriminate between classes — but which do have some diagnostic power may be viewed as the combination of modal features with

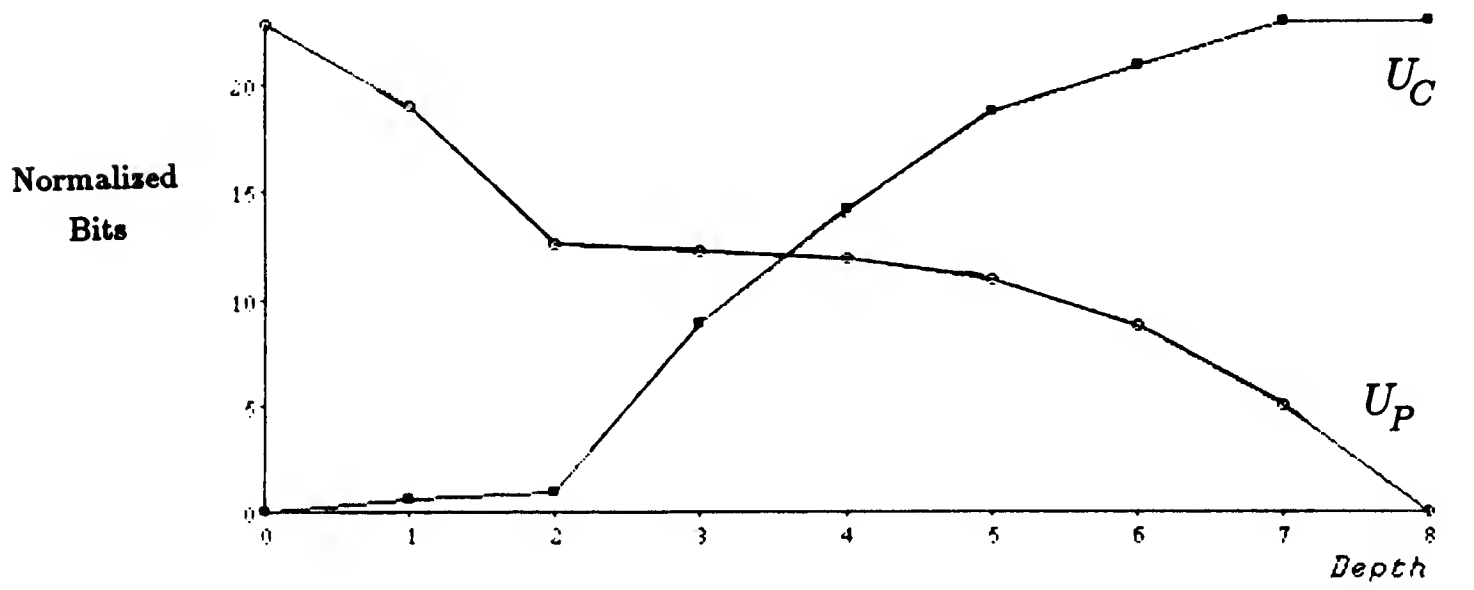


Figure 4.15: The evaluation curves for the ordered taxonomy, but with the addition of four noise features. Both U_P and the normalized U_C show a definite break at level 2 — the level corresponding to the separate species — but the curves are becoming more like those of the jumbled evaluation.

noise features. To further illustrate this point, we can add more noise to the leaf example by including pure noise features. Figure 4.15 displays the results of using the same ordered taxonomy of Figure 4.12 but with the addition of 4 noise features; for each leaf, each of the noise features was randomly assigned one of 4 values. There is still a definite break in both the U_P and the U_C curves, but they are becoming more like those of the jumbled evaluation of Figure 4.13. This graceful degradation with the addition of noise is essential if the category evaluation function is to be included in a robust method for recovering natural categories.

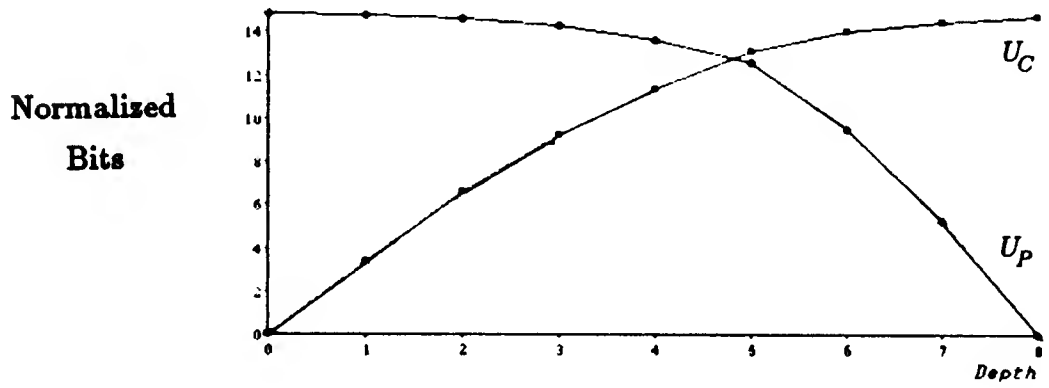
To summarize, we have empirically shown that the evaluation function is indeed sensitive to the structure of natural classes — in this case different leaf species. This sensitivity is indicated by the marked change in the behavior of the quantities U_P and U_C at the depth of the taxonomy which corresponds to the "correct" categories. Also, the components of the evaluation function behave predictably in the absence of natural categories; this last point is crucial since if we are to use this evaluation function to recover

natural categorizations we must be able to distinguish between a minimum caused by structure and a minimum which occurs at some arbitrary level of categorization. To explore this question further let us investigate how the parameter λ affects the evaluation of the taxonomies.

4.6.4 λ-space behavior

Let us return to our task of selecting the best level of a taxonomy for a given λ. Figure 4.16 shows the graphs of the total uncertainty $U = (1 - \lambda)\, U_P + \lambda\, \eta\, U_C$ for the jumbled taxonomy of Figure 4.11 with λ equal to .2, .4, .6, and .8. Selection of the best level for a given lambda is simply finding the depth which has the lowest value of U. In the case of the jumbled taxonomy, only the two extremes of depth are ever the minimum, with the trade-off occurring at about .5. From the graphs for U_P and U_C of Figure 4.16 we can construct the λ-space diagram of Figure 4.17. The complete lack of structure in the taxonomy is reflected in this degenerate λ-space diagram; we have empirically demonstrated the predicted noise behavior of section 4.5.2.

Next let us consider how the total uncertainty U varies with λ for the ordered taxonomy. Graphs of U as a function of taxonomy depth for four different values of λ are shown in Figure 4.18. Notice that for all four values (.2, .4, .6, .8) the second level has the lowest total uncertainty; the second level corresponds to the categorization which contains four categories, each containing all the leaves of one species. Although we know that by design a λ of 0 will select the finest depth (8), and that a λ of 1.0 will select the coarsest depth (0), for this data a λ in the interval of approximately 0.1–0.9 will select the categorization containing four categories. In Figure 4.19 we construct the λ-space diagram for these data. The existence of the large stable region is an indicator that the categorization selected in that region contains categories that are highly structured in terms of the way they minimize the uncertainties of the inferences about an object's properties and its category. It should be noted that the categories selected are those that correspond to the classes of leaves as defined by the botanists.



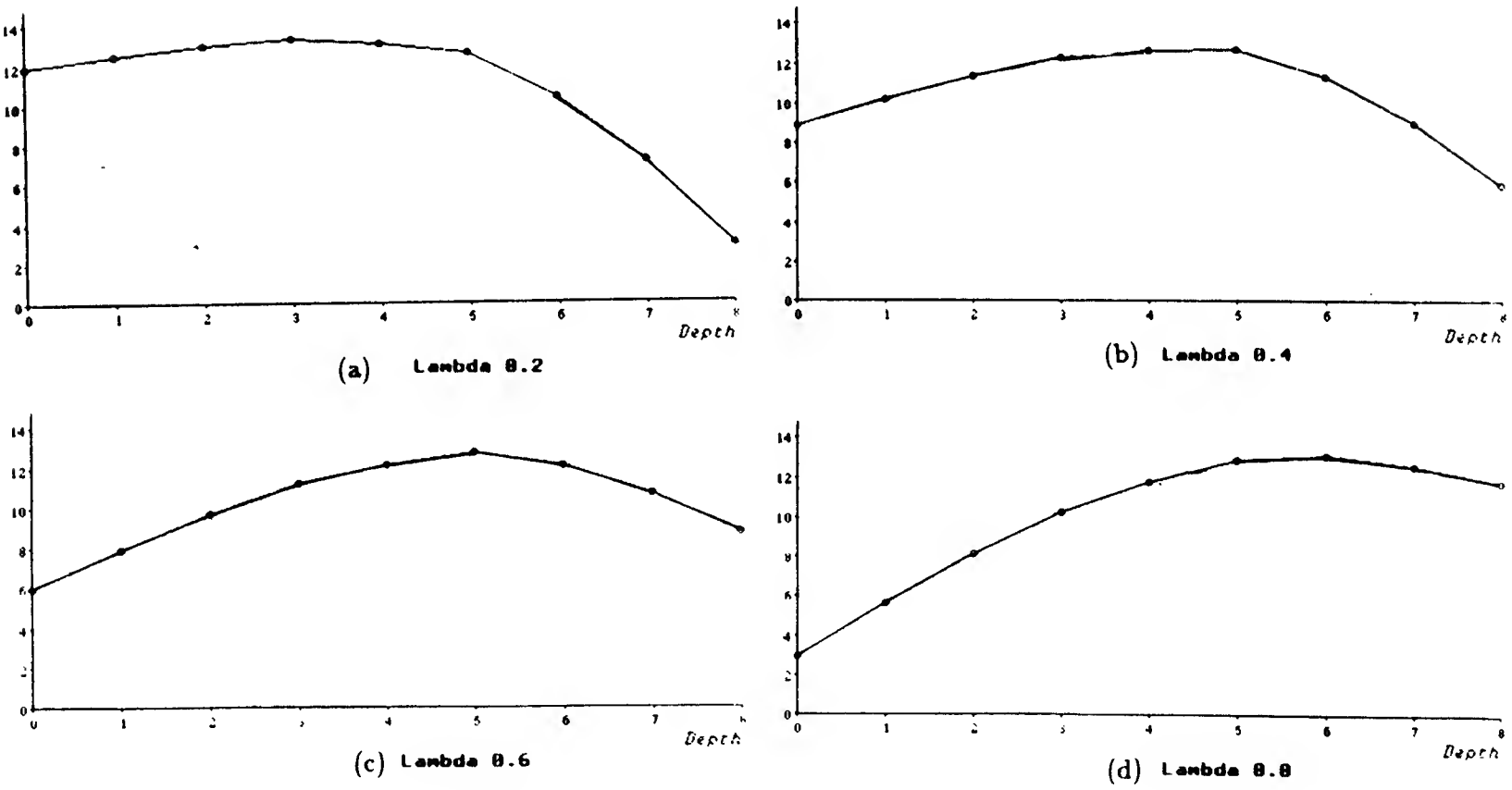


Figure 4.16: Graphs of $U = (1 - \lambda)\, U_P + \lambda\, \eta\, U_C$ for the jumbled taxonomy. Top plot is original vales of U_P and normalized U_C. The four panels a–d are for λ of .2, .4, .6, and .8 respectively.

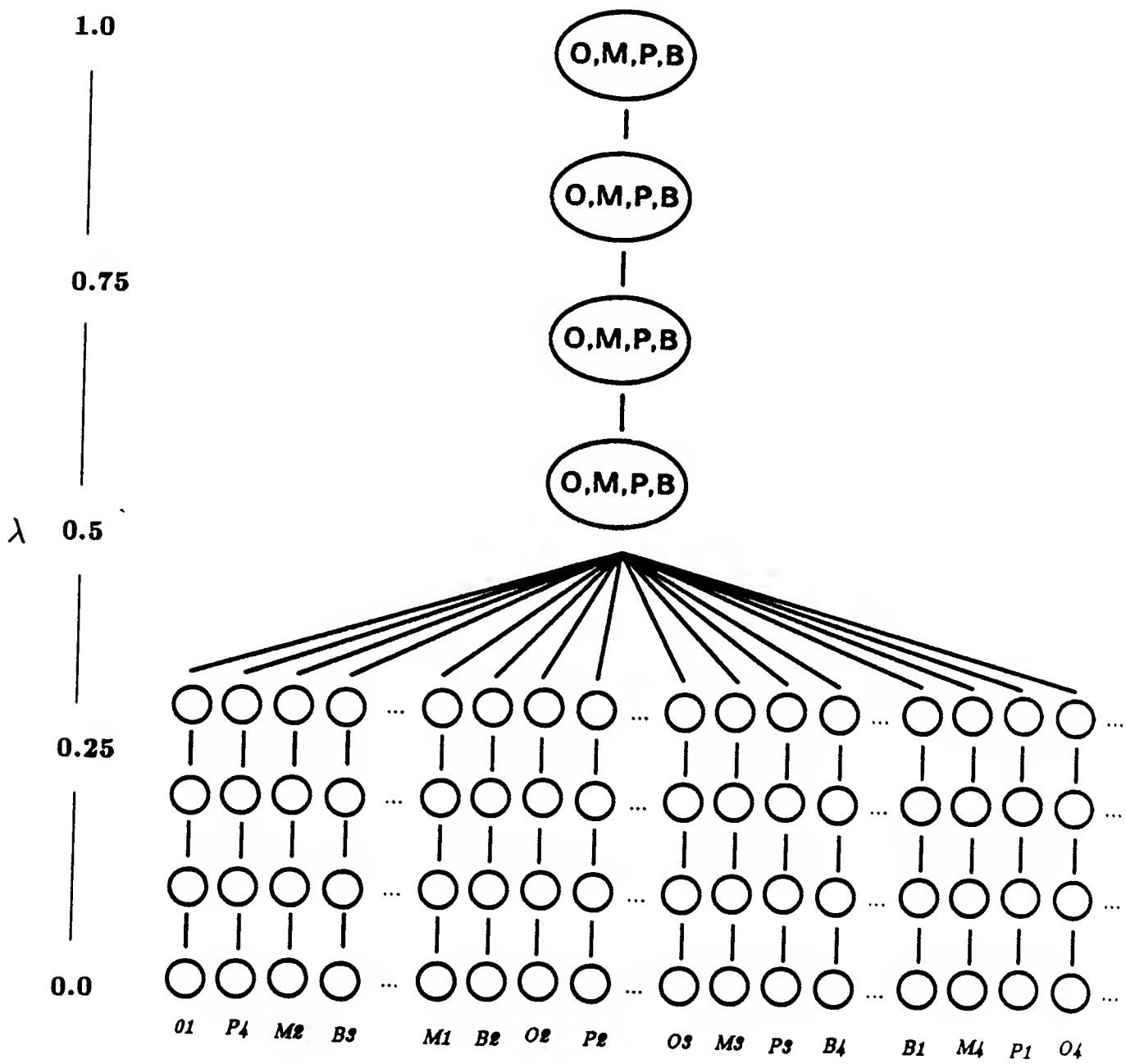
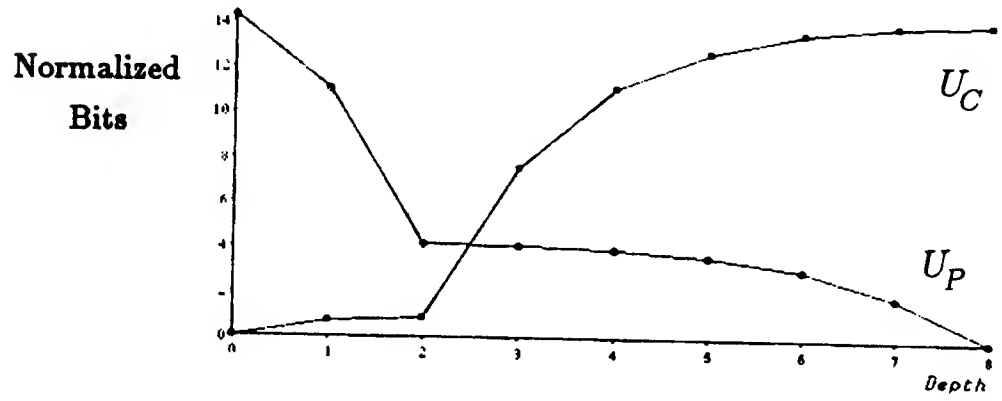


Figure 4.17: The λ-space diagram for the jumbled taxonomy. The degenerate condition of only having only the extreme categorizations be stable reflects the lack of structure in the taxonomy.



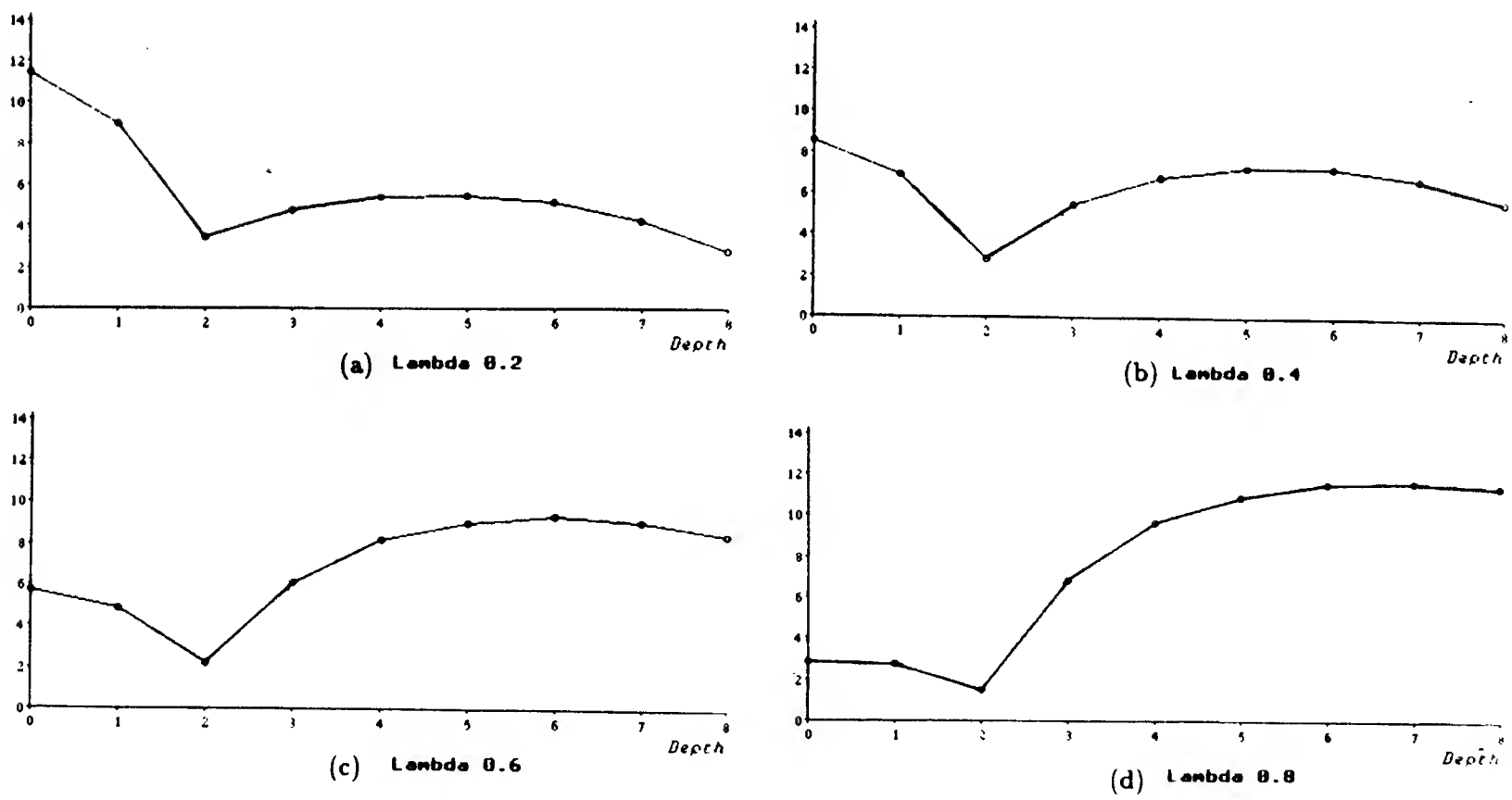


Figure 4.18: Graphs of $U = (1 - \lambda)\, U_P + \lambda\, \eta\, U_C$ for the ordered taxonomy. Top plot is original vales of U_P and normalized U_C. The four panels a–d are for λ of .2, .4, .6, and .8 respectively.

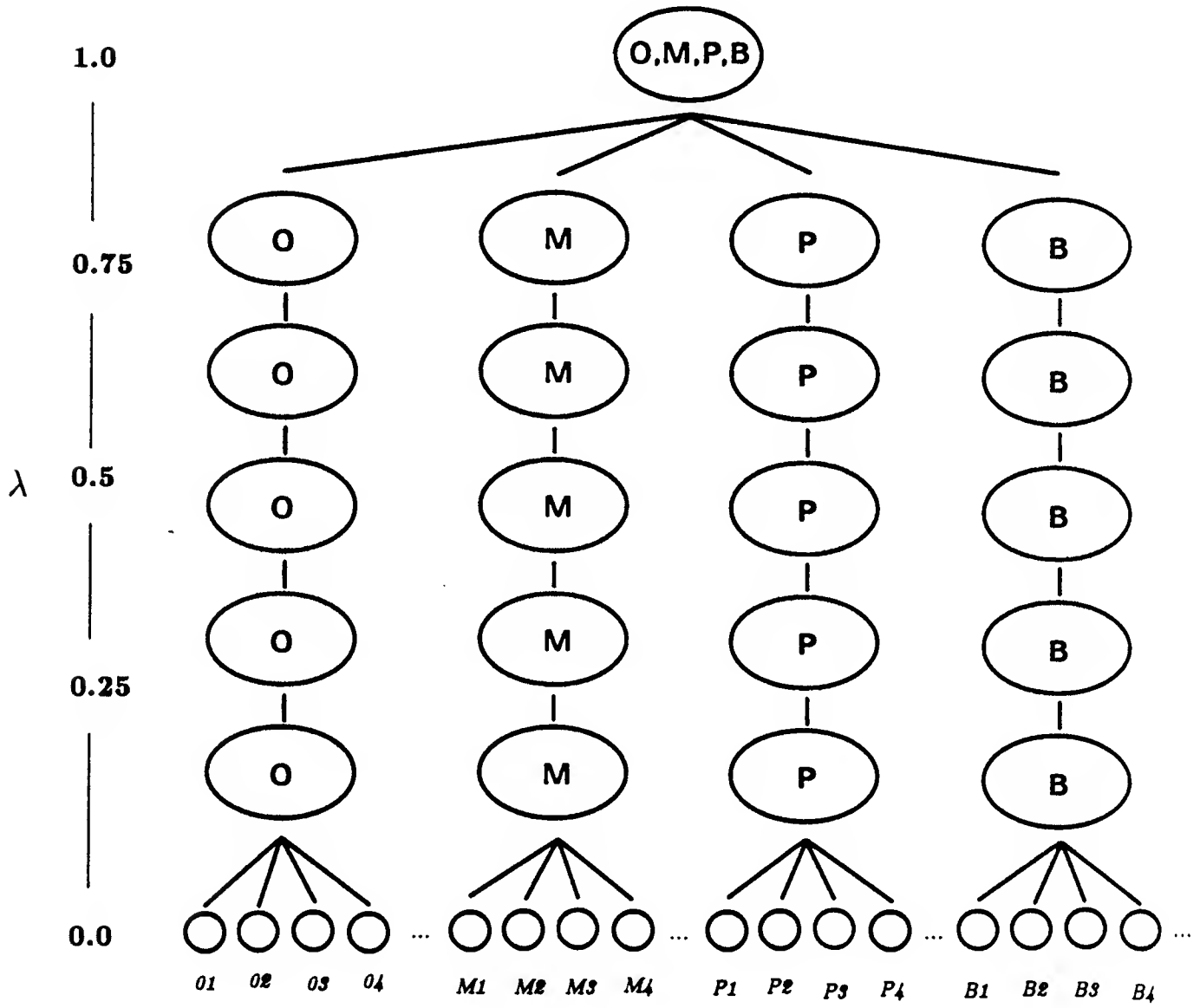


Figure 4.19: The λ-space diagram for the ordered taxonomy. The categorization composed of four categories, each representing one type of leaf, is stable preferred over a wide range of λ, approximately 0.1 to 0.9. This stable region indicates that the four categories are structured in a manner consistent with natural classes.

Chapter 5

Recovering Natural Categories

We have defined the task of the observer to be that of discovering categories of objects in the world that correspond to natural modes; these modes are the natural clusters of objects formed by the interaction between the processes that produce objects and the environmental constraints that act upon them. Because objects of the same natural class behave similarly, establishing a natural categorization — a natural set of categories — permits the observer to make inferences about the properties of an object once the category of that object is determined. The question we address in this chapter is what are the necessary capabilities that must be provided to an observer if he is to accomplish this task?

We can divide the object categorization task into two components. First, if the observer is to ever succeed in generating a natural categorization, then he must be able to determine when a categorization reflects the structure of natural modes. Given a set of alternative categorizations, the observer must be able to select the most likely. Thus, he must be provided with a categorization *evaluation* function. Second, the observer needs a method of producing categorization proposals. As objects are viewed, the observer must be continually refining his current categorization, attempting to recover the natural categories present. The categorization *generation* method must be constructed such that the observer will eventually propose a categorization corresponding to the natural modes.

We begin this chapter by developing a categorization *paradigm* that makes these two components explicit and that agrees with ones intuition about the categorization process; our development of the paradigm is inspired by work

in formal learning theory [Osherson, Stob, and Weinstein, 1986]. Then, we will present a categorization algorithm based upon this paradigm, that has been implemented and tested; the operation and performance of the algorithm is demonstrated by examples drawn from three domains. Analysis of the competence of the algorithm provides insight into the effectiveness of the categorization procedure as well as the types of errors that may be expected. In particular, for certain ideal cases, the algorithm is shown to be guaranteed to converge to the correct categories. Finally, possible modifications of the algorithm to improve its behavior are discussed.

5.1 A Categorization Paradigm

Consider the leaves pictured in Figure 5.1. To most observers there are three groups of leaves present: *ACH, BFG, DEJ*. In fact, botanists would state that there really are three classes of objects present, and that an observer who identifies those three classes has categorized the leaves "correctly." Using these leaves as an example let us develop a paradigm for categorizing objects that is not only consistent with our intuitions about categorization but also permits us to precisely define the object categorization problem.[1] We view the categorization task as a *learning* problem: the observer attempts to learn natural object categories as he inspects the world of objects. Thus, the categorization paradigm we present closely resembles the generalized learning paradigm developed by Osherson, Stob, and Weinstein [1986], based upon the language acquisition paradigm originated by Gold [1967]. Our paradigm consists of four components; each is necessary to define the categorization task precisely.

The first requirement is that the goal of categorization be stated clearly. We define a *categorization* to be a partition of the objects in a population; the equivalence classes of the partition form the categories. Thus, for Figure 5.1, any possible grouping of the leaves constitutes a categorization, and the groups are the categories. However, if the observer is attempting

[1] In Bobick and Richards [1986], a formal description of the categorization paradigm is provided. The terminology developed there permits a formal statement of the categorization problem. However, most of the important issues developed there can be discussed informally, by considering an example problem. The reader is referred to Bobick and Richards [1986] if further detail is required.

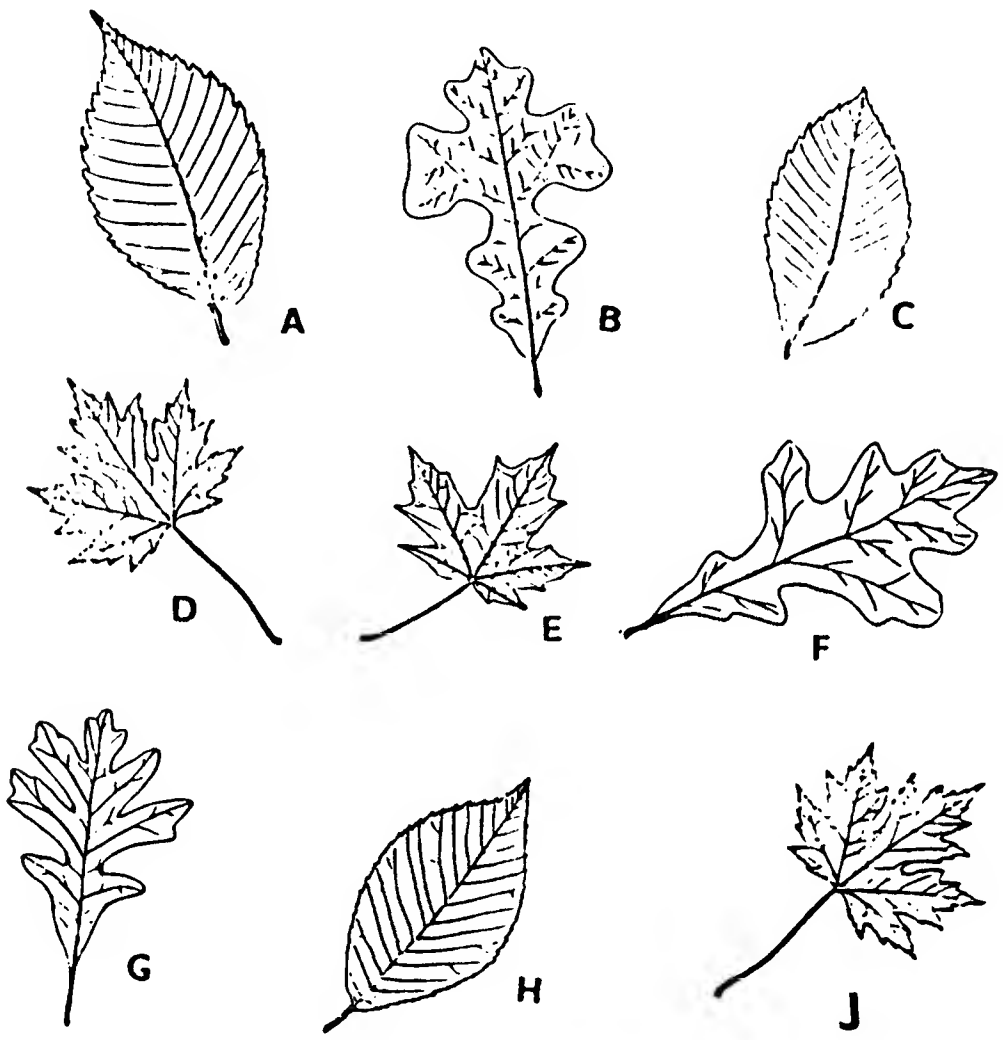


Figure 5.1: A set of 9 leaves. According to the botanists these are 3 instances each of 3 different species of leaves: White Oak, Sugar Maple, Poplar (common names). The categorization problem is to find the natural grouping of these leaves.

to discover the "correct" categories than we need a basis for determining a natural categorization. To provide such a basis we require the Principle of Natural Modes: environmental pressures interacting with natural object processes cause the world to be clustered in the space of properties important to the interaction between objects and the environment. We refer refer to these clusters as *natural classes.* In Figure 5.1 the natural classes correspond to the different species of leaves: White Oak (*BFG*), Sugar Maple (*DEG*), Poplar (*ACH*). The goal of the observer is to recover these natural classes since they represent instances of objects which share common properties; they were generated by the same natural object process. Thus, natural classes serve to define a testable goal of the categorization procedure: produce categories of objects corresponding to natural classes.[2] These *natural categories* are the first component of the categorization paradigm.

Three difficulties arise if we consider the recovery of "natural categories" as an objective goal for the categorization task. Two are philosophical. First, how can we independently judge if the observer is successful? To do so, we require independent identification of the natural classes, an omniscient observer or an oracle. Second, as demonstrated by Goodman [1951], Quine [1969], and Watanabe[1985], natural categories may only be said to exist if we restrict the properties of objects that are considered important. Otherwise, all objects are equally similar. (See section 2.3.1 for a review and proof of Watanabe's Ugly Duckling Theorem, a theorem that explains this counter-intuitive claim.) How then can we say that one set of categories is more natural than another? To resolve these problems, we rely on the sciences that study the domains in question to provide an independent assessment of the natural classes. Because botanists have categorized the leaves in Figure 5.1 into three species, and because botanists study the processes that create leaves and environments that constrains them, we will assume that the categories constructed by botanists represent "true natural classes."

The third difficulty in using natural classes as a baseline against which to judge the competence of the observer is computational in nature. We have

[2] We have purposely avoided using the phrase *the* natural classes because we do not wish to claim that there is a unique clustering of objects corresponding to a natural partitioning. As discussed in chapter 2, both the division between mammals and birds and that between cows and rabbits represent natural clusterings. Thus, two observers could both "correctly" learn the natural categories of the world and yet have different categorizations. We will return to this issue in the next chapter.

defined a categorization to be a partition of the objects in the world. But, there is an uncountable infinity of possible objects.[3] Thus, the number of partitions is also uncountably infinite. Even if there are only denumerably many objects (an assumption that would be valid if objects are produced by a countable number of "computational" construction procedures) there would still be an uncountable set of partitions. How can the observer ever hope to discover the correct set of categories if the space of potential categorizations is unsearchable?

We can remedy this situation by placing a constraint on the *categorization environment* — the third component of the categorization paradigm — and by modifying the goal of observer. When an observer views an object (such as one of the leaves in Figure 5.1), he cannot make use of the object itself as input to a categorization procedure. Rather, he must operate on some sensory description of the objects. Thus, let us construct a categorization environment that consists of objects as described in some *representation.* As in chapter 4, we define a representation to be a mapping from the set of all possible objects onto to some *finite* set Θ^*.[4] Each element of Θ^* is referred to as an *object description.* Because the observer is no longer operating on the objects themselves, but on their description as expressed in some representation, we alter the definition of an object categorization: a categorization $\mathcal{Z}$ is a partition of the set of representational descriptions corresponding to the objects in the world. Now, because there are only a finite number of object descriptions in the representation, the set of possible categorizations is not only countable, but also finite. Thus, one can construct computational procedures capable of searching the space of solutions to the categorization problem.

One may view a representation as a generalization or abstraction mechanism: an infinite number of objects are mapped onto a single point in the representation. Thus an important question arises as to whether a given representation is sufficient to permit correct categorization. Let us (informally) define a *class preserving* representation to be one in which disjoint natural

[3] A quick proof: How many squares are there having an area less than one square foot?

[4] If one is uncomfortable with the concept of "the set of all possible objects" then one can simply define a mapping function, and then let the domain of that function (the scope of the representation as defined by Marr and Nishihara [1978]) become the set of possible objects.

classes of objects in the world map into disjoint sets in Θ^*.[5] If two objects of different natural classes map into the same point in the representation, then the representation is not class preserving and the observer will not be able to correctly categorize the objects. Therefore, for the observer to be successful in his task, the representation must be constrained to match the structure of classes. Once again we encounter the Ugly Duckling Theorem of Watanabe [1986] and require that the representation be chosen so that important properties, in this case those constrained by the natural classes, are made explicit. We refer to the description of classes of objects in terms of a representation as the *projection* of the classes onto that representation.

Having defined a class preserving representation we can now modify the component of the categorization paradigm corresponding to the natural categories. Instead of recovering the natural classes directly, the task of the observer becomes the recovery of the projection of the natural classes in some class preserving representation. The categorization proposed by the observer — the hypothesized partition of representation space — must be constructed such that if two objects in the world are mapped by the representation into the same category (equivalence class) of the partition then those two objects belong to the same natural class.

To complete the definition of the categorization environment, we must specify how the observer comes to experience the objects. In the example of the leaves in Figure 5.1, the observer may simply view all of the objects "simultaneously." For a large or infinite world, a parallel observation of all objects is not possible. Thus we define an *observation sequence* to be an infinite sequence of objects, each described according to some representation; this sequence is viewed serially by the observer. We require that the sequence be infinite so that the observer always has data available as input to a category recovery procedure. However, there are only a finite number of distinct object descriptions. Therefore we will require that any object description that represents some object in the world must appear in the observation sequence an infinite number of times. This property of the observation sequence will be important when we discuss the error correcting capability of a categorization procedure. Note that our definition of observation sequence guarantees that there exists a point in the observation sequence at which the

[5] Bobick and Richards [1986] provides a formal definition of a class preserving representation.

observer will have viewed all the object descriptions representing objects in the world.

After encoding some information about the objects in the world, the observer must propose some candidate categorizations. In our categorization paradigm, we require that the observer announce a categorization hypothesis after each presentation of an object from the observation sequence. We can decompose the task of announcing hypotheses into two components: *hypothesis generation* and *hypothesis evaluation.* These tasks form the last two components of the categorization paradigm.

Hypothesis generation refers to the method used by the observer to propose candidate categorizations. One simple approach would be to simply check all possible partitions of the objects viewed so far: there are only a finite number of object descriptions and thus only a finite number of possible partitions. By our definition of of the observation sequence, we know that the observer only need to wait some finite amount of time before he will have viewed all the object descriptions present in the world. Assuming that the observer's decision criteria — the evaluation procedure to be discussed presently — are capable of selecting the natural class categorization, then an exhaustive enumeration is guaranteed to find the correct categories.

Unfortunately, the combinatorics of an exhaustive search make such a procedure impractical. For the 9 objects of figure Figure 5.1, there are over 20,000 possible partitions. If there are 15 objects, the number of partitions (categorizations) grows to 1.4 billion! Also, an exhaustive partitioning strategy is not well suited to a sequential presentation of objects provided by the categorization environment. When a new object is viewed, the previous hypothesis is irrelevant because an exhaustive search would again be executed and the best partition selected. In a world with thousands of objects, the discarding of previous hypotheses, and the work associated with producing them, is unacceptable.

An alternative to the exhaustive search is a dynamic, data-driven method of hypothesis generation. This is the approach used in dynamic classification (see, for example, Duda and Hart [1973]). At each presentation of an object, the observer considers some (usually small) set of candidate hypotheses based upon the previous hypothesis and the new object. Because one can limit the degree to which any new object may alter the current hypothesis, the incremental strategy has the advantage that the computational complexity of computing the new hypotheses can be restricted.

The use of an incremental approach raises some issues that are not relevant when employing an exhaustive strategy. In particular, one must consider whether the observer will ever converge to some particular hypothesis. Even though we know that there are only a finite number of partitions, it may be the case that the observer never converges to some particular hypothesis; for example, the observer may continually cycle through all the possible partitions. Also, because the observer is not considering all possible hypotheses, we must consider whether he will ever propose the "correct" one. In the next section we will describe an incremental hypothesis generation method that has been successfully demonstrated in several domains. It will be shown that in certain ideal cases, the method can be constructed such that it will converge with unit probability to the "correct" hypothesis; experimental results will demonstrate the method's effectiveness on real data.

Finally, given a set of candidate categorizations, the observer needs to be able to select the one most likely correct: the one which is the most "natural." To accomplish this task, the observer requires a hypothesis *evaluation function.* This function must be constructed such that categories corresponding to the natural classes are preferred over categories that ignore class structure. Like the representation, which is required to make explicit the properties of objects constrained by natural processes, the hypothesis evaluation function must be matched to the structure of the natural world.

Having defined the four components of categorization we may now state the categorization problem more precisely. We assume the following are given: a set of *natural object classes*, a class preserving *representation* in which objects are described, an *observation sequence* in which all the object descriptions are presented, a *hypothesis generation method* to produce candidate categorizations, and a *hypothesis evaluation function* which provides criteria as to which categorization should be chosen. We say that the observer has successfully categorized the world of objects on some observation sequence if and only if 1) he announces some categorization hypothesis after every presentation of an object description, and 2) the observer eventually converges to a hypothesis which is the projection the natural classes in the class preserving representation. By "converge" we mean that the observer eventually announces the correct hypothesis and that he never deviates from that hypothesis as he continues to view the observation sequence.

Notice that any particular categorization of objects is learnable. A strongly

nativist theory of object categorization would claim that the observer always announces a hypothesis corresponding to one particular categorization $\mathcal{Z}_0$, where $\mathcal{Z}_0$ has been selected by evolution to appropriately categorize the world. That is, the observer would completely ignore the data of the observation sequence. However, it is unreasonable to expect evolution to provide for an object category such as "refrigerator." A more plausible theory of categorization, and that which has been proposed here, is that evolution equips the observer with the necessary tools — representation, hypothesis generation method, hypothesis evaluation function — for the recovery of natural object categories.

5.2 Categorization Algorithm

In this section, we will present a categorization system that reflects the paradigm developed above; this system has been implemented and tested. Because the representation (the most important aspect of the categorization environment) and the hypothesis evaluation function are described in detail in chapter 4, we will only provide a brief description of these components of the categorization system. The hypothesis generation method, however, will be presented in detail. We will evaluate the performance of the algorithm by examining the results of tests conducted in three domains. In the following sections, we will consider the competence of the categorization algorithm, the types of errors likely to arise, and possible remedies.

5.2.1 Categorization environment

The representation — the first component of the categorization environment — used by the categorization system consists of *property vectors*. Our terminology is defined as follows: *feature* refers to a function or predicate computed about an object; *value*, to the value taken by a feature; *property*, to the valued feature. For example, "length" is a feature, "6 ft." is a value, and "having length 6 ft." is a property. Each feature, f_i, $1 \leq i \leq m$, has an associated set of values $\{v_{i1}, v_{i2}, \ldots, v_{in}\}$ referred to as the range of the feature. We require that the range be a finite set but the cardinality of the range can vary from one feature to the next. **F** denotes the set of features $\{f_1, f_2, \ldots, f_m\}$ Using these features, each object is represented by an m-

dimensional property vector $\mathbf{P} = (v_{1\alpha}, v_{2\beta}, \ldots, v_{m\gamma})$ where v_{ij} is the j^{th} value of the range of the i^{th} feature.

To complete our specification of the categorization environment requires the generation of an observation sequence. Normally, the world itself provides a set of objects that can be sampled to form the sequence. However, to test the categorization algorithm we need to generate property vectors corresponding to objects. Furthermore, to evaluate the performance of the categorization system these property vectors must be constructed such that a natural categorization exists. To satisfy these criteria, *property specifications* — a listing of acceptable property values — are provided for several classes of objects. An object generator then creates property vectors consistent with these specifications. We will describe the specific features and values of the property specifications when we present the examples illustrating the performance of the categorization procedure in different domains.

5.2.2 Categorization uncertainty as an evaluation function

The hypothesis evaluation function provides the criteria by which proposed categorizations are selected. Because the categorization task requires recovering the natural categories, the evaluation function must reflect the natural structure found in the world.

The evaluation function is based upon the *categorization uncertainty* measure U. It is defined by:

$$U(\mathcal{Z}) = (1 - \lambda)\, U_P(\mathcal{Z}) + \lambda\, \eta(\mathcal{Z})\, U_C(\mathcal{Z}) \tag{5.1}$$

where U_P is the uncertainty about the properties of an object once its category is known, U_C is the average uncertainty of the category to which an object belongs given a subset of the properties describing the object, η is a normalization coefficient between U_P and U_C, and λ is a free parameter representing the desired trade-off between the two uncertainties. (See chapter 4 for complete definitions and derivations of these terms.) For the remainder of this chapter we will assume that λ is set to some particular value which satisfies the goals of the observer. In chapter 6 we will consider the effect of λ on the categorization procedure and its interaction with the natural object categories.

We use the total uncertainty of a categorization U as an evaluation function because this measure reflects the degree to which a categorization permits the observer to accomplish the recognition goal of making reliable inferences about the properties of objects. Thus a categorization which minimizes U is guaranteed to be useful to the observer: *the evaluation function directly measures the utility of a categorization.* This desirable property of the evaluation function is absent in the standard distance metrics employed by cluster analysis techniques. Furthermore, the Principle of Natural Modes supports the claim that if a categorization supports the goals of the observer then that categorization reflects a structuring of the objects consistent with the natural modes. As such, this function is well suited for the evaluation of proposed categorizations.

5.2.3 Hypothesis generation

The hypothesis generation method we present has been designed to be consistent with the categorization paradigm. First, the algorithm is guaranteed to produce a hypothesis at each point along the observation sequence. Thus, the observer will never halt and refuse to announce a categorization. Second, categories are permitted to continually split and merge making every possible categorization fall within the scope of the algorithm. Finally, the algorithm takes advantage of the infinite observation sequence by correcting "mistakes" only when viewing an object previously placed in an incorrect category. Because the observer is guaranteed to view each object repeatedly, this form of data driven error correction is appropriate.

The method may be described as a hybrid of divisive and agglomerative clustering techniques [Duda and Hart, 1973; Hand, 1981]. (See chapter 3 for a discussion of these methods.) The basic steps of the algorithm are as follows:

1. Construct an initial categorization consisting of a single category by randomly selecting a small number of objects from the population.

2. View a new object[6] from the observation sequence.

3. Given a current categorization hypothesis, select the category to which adding the new object results in the best new categorization ("best"

[6]The term "object" refers to an object description in the property space representation.

in terms of lowest total uncertainty U.). Add the new item to the "selected" category.

4. Test if merging the selected category with any other category yields a better categorization. If so, merge the selected category with the best of those, and make the resulting category the new selected category.

5. Delete any objects identical to the new object which were previously categorized into a category different than that which was selected during this iteration.

6. If there are "enough" objects in the selected category, attempt to split the category into two new categories such that a better categorization is achieved.

7. Go to Step 2.

We postpone examining the competence of the algorithm until we present examples of its operation.

5.2.4 Example 1: Leaves

The first domain in which we illustrate the performance of the categorization algorithm is that of leaves, like those in Figure 5.1. For several species of leaves, property specifications were generated according to descriptions provided by Preston [1976]. (Table 5.1) The properties chosen are known to be diagnostic of leaf species and thus are sufficient for the categorization task. Note that for these classes of leaves the representation is class preserving: no property vector can be constructed that satisfies more than species specification. For this example, the free parameter λ of the evaluation function U has been set to a value of 0.6.

Let us trace the categorization process by examining the dynamic output of the program shown in figures 5.2 and 5.3. As each new object is viewed, a new row is added, showing the categorization proposed by the system in response to that new object; the new object is shown on the left. In these examples, the object's names are used to indicate (to the programmer) the true classes to which the leaves belong, e.g. COTTON-145 is a cottonwood leaf. The program, of course, uses only the property vectors of the objects as

	Length	Width	Flare	Lobes	Margin	Apex	Base	Color
Maple	{3,4,5,6}	{3,4,5}	0	5	Entire	Acute	Truncate	Light
Poplar	{1,2,3}	{1,2}	{0,1}	1	Crenate, Serrate	Acute	Rounded	Yellow
Oak	{5,6,7,8,9}	{2,3,4,5}	0	7,9	Entire	Rounded	Cumeate	Light
Birch	{2,3,4,5}	{1,2,3}	0	19	Doubly-Serrate	Acute	Rounded	Dark
Cottonwood	{3,4,5,6}	{2,3,4,5}	2	1	Crenate	Acuminate	Truncate	{Light,Dark, Yellow}
Elm	{4,5,6}	{2,3}	{0,-1}	1	Doubly Serrate	Accuminate	Rounded	Dark

Table 5.1: Leaf specifications for several species of leaves. A leaf generator was designed which created property vectors consistent with the different specifications.

input. The circled numbers to left indicate the significant events that we will discuss. In this example, the population consists of 150 leaves, 25 examples of each of 6 species.

Event 1 is the start of the categorization algorithm. Because step 3 of the algorithm requires a current categorization, we begin with an initial categorization consisting of a small random collection of objects forming one category. In the next section, when we analyze the performance and competence of the hypothesis generation method, we will place bounds on how large this initial category may be.

Event 2 represents viewing a new object, in this case the leaf COTTON-145. Step 3 of the algorithm selects the category to which adding the new leaf produces the best categorization. As there is only one category in the current categorization, COTTON-145 is added to that category. Because there are as yet no other categories, the merging step (4) and the deletion step (5) are skipped. Next, the splitting step (6) is executed. It is important to understand the details of this step because the splitting procedure is the only means by which a new category can be created and thus is most critical in determining the competence of the system.

Because the evaluation function U is statistical in nature, based upon probabilities and information theory, it does not yield reliable results when applied to categories that are too small. We restrict its application by requiring that any category formed by splitting contain some minimum number of objects; for this particular example, a category was required to contain at

Start:	BIRCH-106	BIRCH-112	ELM-60	POPLAR-85	COTTON-137	COTTON-142	COTTON-138
COTTON-145:	COTTON-145	BIRCH-106	BIRCH-112	ELM-60	POPLAR-85	COTTON-137	COTTON-142	COTTON-138
COTTON-148:	COTTON-148	COTTON-137	COTTON-145	COTTON-142	POPLAR-85	ELM-80	BIRCH-108	BIRCH-112	COTTON-138
ELM-51:	COTTON-148	COTTON-137	COTTON-145	COTTON-142	POPLAR-85	ELM-51	ELM-80	BIRCH-106	BIRCH-112	COTTON-136
OAK-37:	COTTON-148	COTTON-137	COTTON-145	COTTON-142	POPLAR-65	OAK-37	ELM-51	ELM-80	BIRCH-106	BIRCH-112	COTTON-138
BIRCH-123:	COTTON-148	COTTON-137	COTTON-145	COTTON-142	POPLAR-85	BIRCH-123	OAK-37	ELM-51	ELM-60	BIRCH-106	BIRCH-112	COTTON-138
ELM-59:	COTTON-148	COTTON-137	COTTON-145	COTTON-142	POPLAR-65	OAK-37	ELM-60	ELM-59	COTTON-138	BIRCH-123	6IRCH-112	ELM-51	BIRCH-108
ELM-57:	COTTON-148	COTTON-137	COTTON-145	COTTON-142	POPLAR-65	ELM-57	OAK-37	ELM-86	ELM-59	COTTON-138	BIRCH-123	BIRCH-112	ELM-51	BIRCH-106

* * *

MAPLE-2:	POPLAR-95	ELM-83	ELM-57	MAPLE-2	OAK-47	COTTON-138	OAK-37	COTTON-148	MAPLE-14
	BIRCH-121	ELM-59	ELM-73	MAPLE-21	MAPLE-15	OAK-49	COTTON-144	COTTON-137	POPLAR-85
	BIRCH-117	ELM-89	MAPLE-17	OAK-48	OAK-26			COTTON-142	POPLAR-86
	POPLAR-96	ELM-68	ELM-58	ELM-80	OAK-45			COTTON-136	COTTON-145
	POPLAR-90								
	BIRCH-123								
	BIRCH-112	ELM-51							
	BIRCH-106								
MAPLE-4:	POPLAR-95	ELM-83	ELM-57	COTTON-138	COTTON-148	MAPLE-14	MAPLE-2	OAK-45	OAK-46
	BIRCH-121	ELM-59	ELM-73	OAK-37	OAK-49	COTTON-137	POPLAR-65	MAPLE-21	OAK-26	OAK-47
	BIRCH-117	ELM-89		COTTON-144	COTTON-142	POPLAR-68	MAPLE-4	
	POPLAR-96	MAPLE-17			COTTON-136	COTTON-145	MAPLE-15	
	ELM-68	ELM-56	ELM-80					
	POPLAR-90							
	BIRCH-123							
	BIRCH-112							
	ELM-51							
	BIRCH-106							

* * *

POPLAR-80:	ELM-63	ELM-57	COTTON-138	OAK-37	COTTON-148	MAPLE-2	MAPLE-21	OAK-35	OAK-41	POPLAR-80
	ELM-59	ELM-73	OAK-49	COTTON-144	COTTON-137	MAPLE-4	MAPLE-15	OAK-45	OAK-46	POPLAR-83
	ELM-89	MAPLE-17			COTTON-142			OAK-28	OAK-47	MAPLE-14
	ELM-58	ELM-88			COTTON-136					POPLAR-85
										POPLAR-86
										COTTON-145
										POPLAR-95
										BIRCH-121
										BIRCH-117
										POPLAR-96 ELM-68
										POPLAR-90
										BIRCH-123
										BIRCH-112 ELM-51
										BIRCH-108
OAK-40:	ELM-83	ELM-57	ELM-59	COTTON-148	MAPLE-2	MAPLE-21	POPLAR-60	POPLAR-83	OAK-40	OAK-35	OAK-41
	ELM-73	ELM-89		COTTON-137	MAPLE-4	MAPLE-15	MAPLE-14	POPLAR-85	OAK-45	OAK-48	OAK-28
	MAPLE-17	ELM-56		COTTON-142			POPLAR-68	COTTON-145	OAK-47	COTTON-138
	ELM-60			COTTON-136			POPLAR-95	BIRCH-121	OAK-37	OAK-49
							BIRCH-117	POPLAR-96	COTTON-144	
							ELM-66	POPLAR-90		
							6IRCH-123	BIRCH-112		
							ELM-51	BIRCH-108		

* * *

Figure 5.2: The output of the categorization program. Each row is the proposed categorization after having viewed the object listed at the left. The output continues to the next page.

```
                                                
```

least 4 objects. Therefore the algorithm does not attempt to split a category unless it contains at least twice the minimum number of objects (8 in this example).

Assuming a category is large enough to be split, as is the case at event 2, candidate partitions must be created. Because the number of partitions of a category is huge (even when partitioning a set into only two subsets), not all partitions of a category into two new categories may be attempted. Therefore, only some (randomly chosen) divisions are tried. Thus, if there exists a split of a category that yields a better categorization than the current hypothesis, the probability of discovering that partition is proportional to the number of partitions attempted. In the current implementation the number of partitions considered is proportional to the size of the category; when we discuss the convergence and correctness properties of this algorithm, this sampling rate will become important. At event 2, none of the partitions considered yielded a categorization with a lower uncertainty than the categorization consisting of only one category.

At event 3, however, a partition is accepted. As before, the new leaf (COTTON-148) is added to the only category in the current categorization. However, in this case a split of that category was discovered which yielded a better categorization than the single category. Event 4 is another instance of successful splitting.

One of the dangers of an algorithm such as this is that it is possible to cause two categories to be created which should be one. Event 5 is an example of such an occurrence. In this case, the addition of the leaf MAPLE-4, caused a category to split, separating maples from oaks. But a previous split had already created a category containing oaks. If the algorithm is to successfully categorize these objects, then these two categories must eventually be merged. That is the purpose of step 4 in the algorithm. At event 6, the leaf OAK-40 was initially added to the category with the 6 oak leaves. This category was then merged with the category containing the two other oak leaves. Even though this second category contained two leaves that are not oak, the merging of the two categories yielded a better categorization. Merging assures that splinter categories that are created because of the order of presentation of the objects may later be reclaimed.

One more form of error correction is necessary. Although merging can combine categories mistakenly separated, it cannot remove isolated errors that result from previous mistakes. To correct this type of error, we add

the deletion step of the algorithm (5). Examples of this step are shown at events 7 and 8. At event 7, the leaf COTTON-138 was viewed for the second time, and placed in the category containing only other cottonwood leaves. Notice that there were two cottonwood leaves present in the oak category, one of which was previous instance of COTTON-138. Because the current instance was placed in a different category, the program can correct an earlier "mistake"' by deleting of the previous instance. Event 8 is a similar event where MAPLE-17 was corrected.[7]

The last categorization shown in Figure 5.3 represents the steady state categorization produced by the algorithm; at this point the program was interrupted. Notice that the categorization procedure recovered the natural classes, except for the one category consisting of poplar and birch leaves. Thus, the algorithm converged, although not quite correctly.

The first observation to be made is that the algorithm performs quite well. Most tests performed on the leaves domain yielded results as good as those shown, or better, where the solution was exactly the (botanically) correct categorization. The fact that an evaluation function based upon the goals of an observer and an incremental hypothesis generation method could produce a natural and correct categorization provides empirical support for the categorization principles embodied in the procedure.

However, as shown, the algorithm does make errors, even in a domain where it sometimes generates the correct solution. After presenting another example domain, we will discuss the competence and behavior of the algorithm, the predictable errors, and possible remedies.

5.2.5 Example 2: Bacteria

To further illustrate the behavior of the categorization algorithm, we test the procedure on a domain comprised of infectious bacteria. For these examples, property specifications for six different species of bacteria were encoded. Table 5.2 displays the specifications for these species; the data are taken from [Dowell and Allen, 1981]. Because most of the "real" features take on only one value per species (unlike the leaves where features like "length"

[7] A note about the implementation: Because the categorization evaluation function requires that the categories be sufficiently large, categories that grow too small because of this deletion step are themselves deleted. The infinite observation sequence guarantees that these objects will be viewed again.

	BF *bacteroides fragilis*	BT *bacteroides thetaiotamicron*	BV *bacteroides vulgatus*	FM *fusobacterium mortiferum*	FN *fusobacterium necrophorum*	FV *fusobacterium varium*
loc	GI	GI	GI	OR	OR	OR
gram	neg	neg	neg	neg	neg	neg
gr-pen	R	R	R	{R,S}	S	{R,S}
gr-rif	S	S	S	R	S	R
gr-kan	R	R	R	S	S	S
dole	neg	pos	neg	neg	pos	pos
esculin	pos	pos	neg	pos	neg	neg
bile	E	E	E	E	I	E
glc	ls	ls	ls	none	none	none
rham	neg	pos	pos	{neg,pos}	{neg,pos}	{neg,pos}
nf1	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}
nf2	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}

Table 5.2: The property specifications for six species of bacteria. Because most of the real features have only one value (unlike the leaves where features like "length" and "width" varied greatly) two noise features are added (**nf1** and **nf2**).

and "width" varied greatly) two noise features were added (**nf1** and **nf2**). These features prevented all objects of the same class from having identical property descriptions.

Of the six species, three are from the genus *bacteroides*; these are abbreviated as *BF*, *BT*, and *BV*. The other three — *FM*, *FN*, and *FV* — are from the genus *fusobacterium*. Notice that several of the features of the specifications are determined by the genus, while others are determined by the species. For example, all members of *bacteroides* have the property "gr-kan = R" (coding for "growth in presence of Kanamycin is resistant"). Other properties, such as "dole," vary between the species, ignoring genus boundaries. These data were chosen as an example of a population in which there is more than one natural clustering. In this chapter we are only concerned illustrating the operation of the categorization algorithm. In the next chapter we consider the issue of multiple clusterings, and the interaction between λ and the categories recovered.

Figure 5.4 displays the results of executing the categorization procedure with λ set to 0.65. Notice that the bacteria have been categorized according to their genus. Is this the "correct" solution? As mentioned in chapters 2 and

Start:	FV-72 BV-30 BF-1 FN-53 FM-45 BF-7 BT-21		
FN-51:	FN-51 FM-45 FN-53 FV-72	BT-21 BF-1 BF-7 BV-30	
BT-23:	FN-51 FM-45 FN-53 FV-72	BT-23 BT-21 BF-1 BF-7 BV-30	
FV-61:	FV-61 FN-51 FM-45 FN-53 FV-72	BT-23 BT-21 BF-1 BF-7 BV-30	
FN-59:	FN-59 FV-61 FN-51 FM-45 FN-53 FV-72	BT-23 BT-21 BF-1 BF-7 BV-30	
FM-48:	FM-48 FN-59 FV-61 FN-51 FM-45 FN-53 FV-72	BT-23 BT-21 BF-1 BF-7 BV-30	
FM-44:	BT-23 BT-21 BF-1 BF-7 BV-30	FV-72 FN-51 FN-59 FN-53	FM-44 FM-48 FV-61 FM-45
FN-58:	BT-23 BT-21 BF-1 BF-7 BV-30	FN-58 FV-72 FN-51 FN-59 FN-53	FM-44 FM-48 FV-61 FM-45
BV-35:	BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FN-58 FV-72 FN-51 FN-59 FN-53	FM-44 FM-48 FV-61 FM-45
BF-4:	BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FN-58 FV-72 FN-51 FN-59 FN-53	FM-44 FM-48 FV-61 FM-45
FV-64:	BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	FM-44 FM-48 FV-61 FM-45
BV-36:	BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	FM-44 FM-48 FV-61 FM-45

* * * *

BT-15:	BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	FM-41 FM-46 FM-44 FM-48 FV-61 FM-45
BV-27:	BV-27 BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	FM-41 FM-46 FM-44 FM-48 FV-61 FM-45

* * * *

BV-28:	BV-28 BT-22 BF-11 BV-34 BV-26 BV-27 BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-65 FV-69 FN-49 FV-63 FV-68 FN-54 FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	FM-37 FM-41 FM-46 FM-44 FM-48 FV-61 FM-45
FV-70:	BV-28 BT-22 BF-11 BV-34 BV-26 BV-27 BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-70 FV-65 FV-69 FN-49 FV-63 FV-68 FN-54 FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	FM-37 FM-41 FM-46 FM-44 FM-48 FV-61 FM-45
FM-40:	BV-28 BT-22 BF-11 BV-34 BV-26 BV-27 BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FM-40 FM-37 FM-41 FM-46 FM-44 FM-48 FV-61 FM-45 FV-70 FV-65 FV-69 FN-49 FV-63 FV-68 FN-54 FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	
BT-20:	BT-20 BV-28 BT-22 BF-11 BV-34 BV-26 BV-27 BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FM-40 FM-37 FM-41 FM-46 FM-44 FM-48 FV-61 FM-45 FV-70 FV-65 FV-69 FN-49 FV-63 FV-68 FN-54 FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	
FV-66:	BT-20 BV-28 BT-22 BF-11 BV-34 BV-26 BV-27 BT-15 BF-9 BT-16 BT-17 BT-13 BF-12 BV-31 BV-25 BV-36 BF-4 BV-35 BT-23 BT-21 BF-1 BF-7 BV-30	FV-66 FM-40 FM-37 FM-41 FM-46 FM-44 FM-48 FV-61 FM-45 FV-70 FV-65 FV-69 FN-49 FV-63 FV-68 FN-54 FV-71 FN-55 FV-62 FV-64 FN-58 FV-72 FN-51 FN-59 FN-53	

Figure 5.4: Categorizing bacteria. In this example λ equals 0.65. The categories recovered correspond to the different genera.

4, the natural clustering of objects in the world occurs at many levels. Mammals and birds represent one natural clustering; cows and horses, another. That the categorization procedure recovered the different genera is another demonstration of the ability of the algorithm to recover natural categories. These two genera consist of two distinct types of bacteria: the *bacteroides* are only found in the GI tract and the *fusobacterium* are located in the oral cavity. Thus, the categorization recovered in Figure 5.4 is *a* correct solution.

5.2.6 Example 3: Soybean diseases

The last domain in which we demonstrate the effectiveness of the categorization algorithm is that of soybean plant diseases. These data are of interest because they have been used by previous researchers to demonstrate the competence of clustering algorithms. Michalski and Stepp [1983b] make use of these data to demonstrate the effectiveness of their conceptual clustering technique (see discussion in chapter 3); at the same time they demonstrate that several standard numerical clustering techniques are *not* capable of recovering the correct categories. Thus, these data provide a means by which to measure the performance of the categorization algorithm relative to other clustering procedures.

Table 5.3 displays the property specifications for each of four different soybean plant diseases;[8] these data are derived from the data presented in Stepp [1983]. In their original form, the data were listed simply as property vectors of several instances. In order to provide a population large enough for the application of the categorization algorithm, a property specification for each species was derived by taking the union of the values of the features for all instances of that species. For example, the "time" feature for disease *Rhizoctonia Root Rot* has the specifed values of {3,4,5,6}; thus, each of these values occurred in at least one property vector for an instance of that disease.[9] Notice that these properties contain much more noise and are less modal than either of the two previous examples of leaves and bacteria. Successful

[8]The letters A, B, C, and D of the top line are used for display in the program output.

[9]Another modification was the deletion of constant features — features that took the same value for all instances. In chapter 6 we show that constant features have no effect on the categorization uncertainty measure U and thus can be removed from consideration. Removing extra features reduces the number of feature subsets and makes the categorization algorithm more efficient.

categorization of this domain requires that the algorithm be insensitive to unconstrained features and robust in its category evaluation.

Figure 5.5 displays the results of executing the categorization algorithm with a λ of .5. Notice that the correct categories, those corresponding to the species, have been recovered. We should emphasize that the algorithm is *not* told how many categories are present, unlike that of Michalski and Stepp [1983b]. Rather, the algorithm discovers the appropriate number of classes in its search for natural categories.[10] The fact that the categorization procedure is capable of recovering the correct categories in this complex domain – a domain in which other clustering techniques have failed — validates the algorithm as a useful categorization technique.

5.3 Categorization competence

We have demonstrated the effectiveness of the categorization algorithm in several domains. However, as shown in the leaves example, the algorithm does not always converge to the correct solution, even in a domain where it *sometimes* does produce the correct categorization. To understand the behavior of the categorization procedure we need to analyze the *competence* of the algorithm. The case we consider is when there are only two classes of objects in a population. The study of this problem will also provide insight into the behavior of the algorithm when there are more classes present. We assume that the representation is class preserving, making the categorization task possible. The issue is whether the algorithm will recover two categories corresponding to the two classes.

Because we start with a categorization consisting of one category, the

[10]To be complete we should mention how the categorization of the soybean diseases varies as we change λ; the value of λ can affect the categories that are recovered. In fact, unlike the leaves or the bacteria example, there does exists another categorization that is reliably recovered by the categorization algorithm. When the value of λ is .65, the recovered categorization consists of the three categories A, B, and {C,D}. This situation indicates that there are *two* natural levels of categorization in this domain. In chapter 6 we explore the issue of multiple modal levels, where more than one level of constraint is operating in a population. Our primary example in that chapter will be the bacteria where the genera and the species provide multiple levels of constraint. However, because we do not have any objective evidence of multiple levels within the soybean domain, we present the multiple categorizations of the soybean diseases in Appendix B.

	A *Diaporthe Stem Canker*	B *Charcoal Rot*	D *Rhizoctonia Root Rot*	D *Phytophthora Rot*
time	{3,4,5,6}	{3,4,5,6}	{0,2,3,4}	{0,1,2,3}
stand	0	0	{1,0}	1
precip	2	0	2	2
temp	1	{1,2}	0	{0,1}
hail	0	{0,1}	{0,1}	0
years	{1,2,3}	{0,1,2,3}	{0,1,2,3}	{0,1,2,3}
damage	{0,1}	{2,3}	1	1
severity	{1,2}	1	{1,2}	{1,2}
treatment	{0,1}	{0,1}	{0,1}	{0,1}
germ	{0,1,2}	{0,1,2}	{1,2}	{0,1,2}
height	1	1	1	1
cond	1	1	0	1
lodging	{0,1}	{0,1}	0	0
cankers	3	0	1	{1,2}
color	{0,1}	3	1	2
fruit	1	0	0	0
decay	1	0	1	{0,1}
mycelium	0	0	{0,1}	0
intern	0	2	0	0
sclerotia	0	1	0	0
pod	0	0	3	3
root	0	0	0	1

Table 5.3: The property specifications for four species of soybean plant diseases. These data are derived from Stepp [1985].

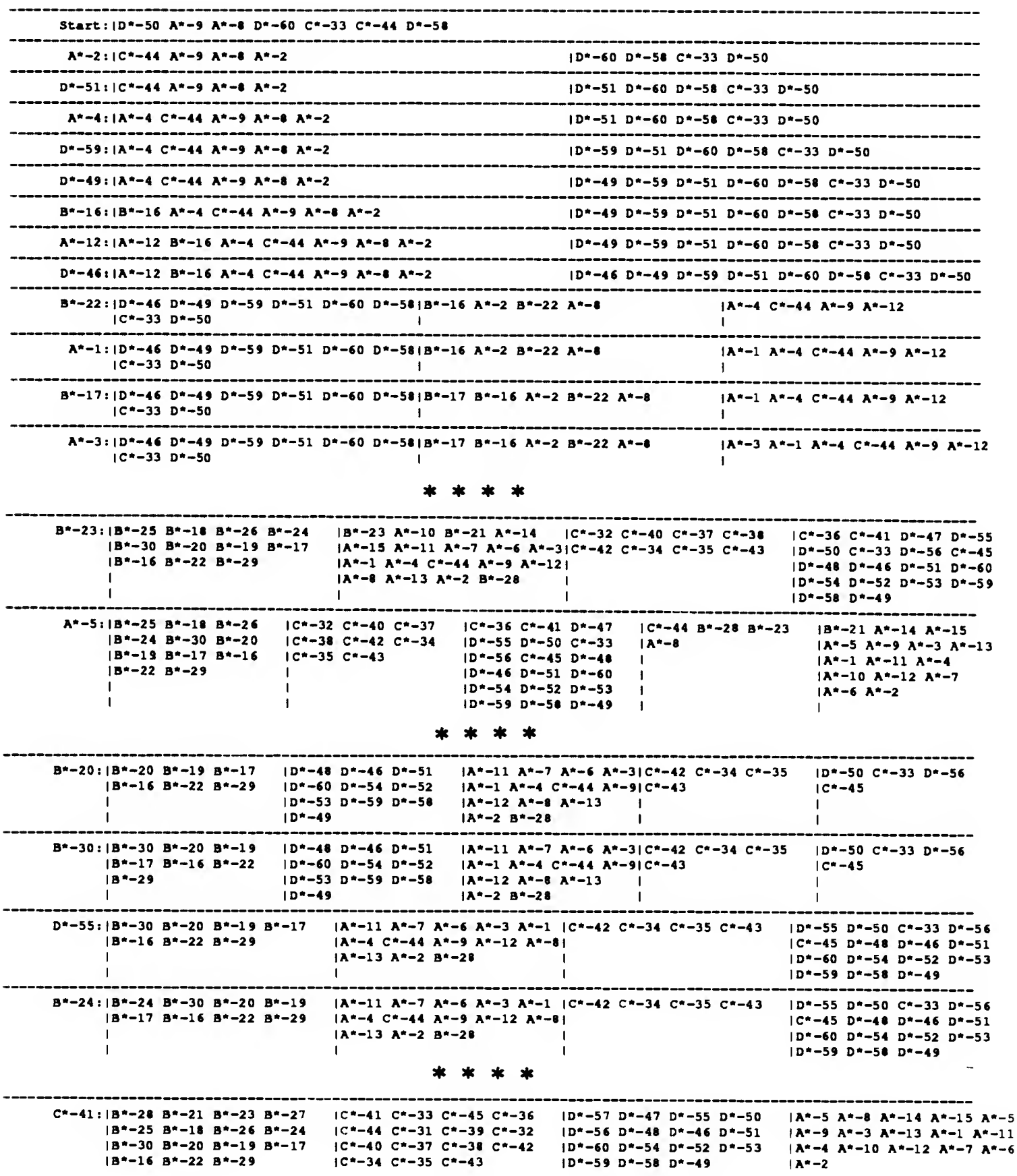


Figure 5.5: Execution of the categorization algorithm in the domain of soybean plant diseases described in Table 5.3. For this example, the value of λ is .5. The categorization algorithm successfully recovers the species.

answer to the above question depends upon whether the algorithm will ever split a category containing both classes into two categories approximating the natural classes. By "approximating" we mean that the two new categories formed are such that any new objects seen will be categorized according to their true classes. Any mistakes caused by an inexact split would be corrected when the objects are viewed again at a later time; because the observation sequence contains an infinite number of repetitions of each object description we know that every object will be encountered again. We refer to these approximate categories as *captivating* categories: they capture all new objects presented that are members of the appropriate class. Notice that whether a category is captivating depends upon the hypothesis evaluation function (in this case the uncertainty measure U) as well as the other categories present: the evaluation function determines to which category an object is assigned.

For the moment, let us suppose that the only partition of the category containing two classes that yields a better categorization than the original is the partition that exactly separates the two classes. We wish to know whether that one partition will be discovered. We assume that the original category is formed by viewing an observation sequence which is unbiased in its distribution of objects from the two classes. Thus, we assume that we have a current categorization consisting of one category of size $2n$ and that contained in the single category are n objects of each of the two classes.[11] We also assume that only partitions of two equal sized categories are considered as possible split categorizations; there are $\binom{2n}{n}/2$ such partitions. If the hypothesis generation method proposes only a fixed number of partitions at each iteration, then the probability of finding the one categorization corresponding to the correct categories rapidly decreases as the single category size ($2n$) increases. Because $\binom{2n}{n}/2 \gg 2^n$, the probability of finding the correct split during the current iteration quickly vanishes to zero as new objects are added to the single category. The exponential rate of increase in the number of partitions guarantees this is true even if the number of partitions considered increases polynomially with the size of the single category. (In the current implementation the number of partitions considered increased linearly with the size of the category.) Thus, having a high probability of discovering the correct class-separating categorization requires doing so be-

[11] At the conclusion of the analysis we will briefly consider the case where the single category contains an unequal number of objects from the two classes.

fore the single category becomes "too large."

To further refine our analysis of the combinatorics of category formation, it is necessary to make some stronger assumptions about the objects being categorized. Let us consider the case of a *modal world.* Here, the object classes and features are such that every feature takes on a different value for each different class. Let F be the number of features in the current representation. We continue to assume that there are only two natural classes present; thus each feature takes on only two values over the entire population. The uncertainty measure U has been constructed such that in a modal world, the best possible categorization was that which separated the modal classes. The question we wish to consider is what is the probability that the categorization procedure presented above will indeed separate the two classes present.

Let us define a k-overlap partition of a two-class category as a partition where k objects of each class are in the wrong category. That is, both categories of the partition contain $n - k$ objects of one class (called the primary class) and k of the other. Thus, $k \leq (n/2)$. Because each category contains k objects that are members of a class of which there are a total of n objects, there are $\binom{n}{k}^2$ k-overlap partitions of a category of $2n$ objects.[12]

The importance of k-overlap partitions is their role in following proposition, whose proof we postpone until we derive analytic expression for the total uncertainty U of a k-overlap partition:

Proposition 5.1 *In a modal world, the categories of a k-overlap partition are captivating for their primary classes if* $k < (n/2)$ *and if the uncertainty measure* U *is used as the hypothesis evaluation function.*

This proposition implies that the creation of a k-overlap partition of a modal world is sufficient to guarantee that the modal classes will be recovered by the categorization procedure: once the categories become captivating, all new objects are categorized correctly and previous mistakes are corrected when the incorrectly categorized objects are viewed again. Thus the probability of correctly categorizing the modal world is equivalent to the probability that a k-overlap partition is created by the splitting step of the categorization procedure. To determine this probability we first need to derive an expres-

[12] If $k = (\frac{n}{2})$ then the number of k-overlap partitions is $\binom{n}{k}^2/2$.

sion for the total uncertainty of a k-overlap partition relative to the total uncertainty of the single category categorization.

The total uncertainty for the single category categorization is simple to derive. Because there is only one category, there is no category uncertainty: $U_C = 0$. Because each of the F features takes on only two values, and because they are evenly distributed, the property uncertainty U_P is equal to $F \cdot (-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2}) = F$. Thus the total uncertainty U of the single category categorization is $(1-\lambda)F$.

The expressions for the k-overlap partition of the modal, two-class world are more complicated. First, because U_C is non-zero when $k \neq 0$, we need to compute the normalization factor η — the ratio between the property uncertainty of the coarsest categorization (a single category) and the category uncertainty of the finest categorization. We have already shown that $U_P(\mathit{Coarsest}(\mathcal{Z}))$ is equal to F. It is easy to show that $U_C(\mathit{Finest}(\mathcal{Z})) = \log n$: If each object is its own category, then in this two class, modal world there are two sets of n identical categories. Because objects of modal classes have identical properties, all subsets of features are equally diagnostic; each subset is consistent with membership in n categories.[13] Thus the category uncertainty is $\log n$ for each subset of features, yielding an average category uncertainty of the same value: $U_C = \log n$.

To develop the expressions for U_P and U_C for a k-overlap partition, we will explicitly derive them for the case $k = 1$. The case $k > 1$ follows analogously. First, consider the property uncertainty of the 1-overlap partition. For each feature, there are two values present in each category; one value (that of the primary class objects) occurs $n-1$ times, the other value, only once. Thus the property uncertainty (for each category, and therefore for their average) is given by:

$$U_P{}^1 = F \cdot \left[-\frac{1}{n}\log\left(\frac{1}{n}\right) - \frac{n-1}{n}\log\left(\frac{n-1}{n}\right) \right] \tag{5.2}$$

or

$$U_P{}^1 = F \cdot \left[\frac{1}{n}\log\left(\frac{n}{1}\right) + \frac{n-1}{n}\log\left(\frac{n}{n-1}\right) \right] \tag{5.3}$$

[13] For the analysis presented here, we assume that the null set is not an allowed subset of features. Otherwise, for that one subset, the number of categories consistent with the (null) property description would be $2n$.

The derivation of U_C is similar. Because we are considering a 1-overlap partition of a modal world, any subset of the properties of an object (as before we exclude the null set) is consistent with $n-1$ objects in one category (the correct "category") and 1 object in the other. Thus the category uncertainty remains constant for all objects and for all subsets. Its value (and also the value of U_C for the 1-partition, because U_C is simply the average of this constant value) is given by:

$$U_C{}^1 = \left[\frac{1}{n}\log\left(\frac{n}{1}\right) + \frac{n-1}{n}\log\left(\frac{n}{n-1}\right)\right] \tag{5.4}$$

The argument is the same for the general k-overlap partition, where $k \leq n/2$. The complete expressions are:

$$U_P{}^k = F\cdot\left[\frac{k}{n}\log\left(\frac{n}{k}\right) + \frac{n-k}{n}\log\left(\frac{n}{n-k}\right)\right] \tag{5.5}$$

$$U_C{}^k = \left[\frac{k}{n}\log\left(\frac{n}{k}\right) + \frac{n-k}{n}\log\left(\frac{n}{n-k}\right)\right] \tag{5.6}$$

Two properties of the above expressions are worth noting. First, $U_P = U_C = 0$ when $k = 0$.[14] This situation is equivalent to an exact partition of the single category into the two modal classes. By design, the total uncertainty of a categorization that separates modal classes is zero.

Second, by taking the derivative of the above expressions with respect to k, one can show that both of the above quantities achieve a maximum at $k = n/2$; also, the total uncertainty increases monotonically as k increases from zero to $n/2$. Thus, unsurprisingly, the total uncertainty of a k-overlap partition increases as the class overlap of the categories increases and is maximized when exactly half the objects of each class are contained in each of the two categories. However, stated in a different form, this result becomes important. Suppose we have a k-overlap partition where $k < n/2$; the inequality assures that each category has a primary class to which it corresponds. Next, suppose we view a new object. The fact that the total uncertainty increases as the degree of overlap increases implies that if we add that object to the incorrect category — the category whose primary class does not correspond to the class from which the new object was drawn

[14] More precisely the $\lim_{k\to 0} k\log\left(\frac{1}{k}\right) = 0$.

— then we will increase the total uncertainty U. Similarly, adding the new object to the correct category will reduce total uncertainty. Therefore, to minimize total uncertainty we will always add new objects to the category corresponding to their primary class. As such, we have now proven proposition 5.1: the categories of a k-overlap partition of a modal world form a set of captivating categories if $k < n/2$ and if the total uncertainty measure U is used as the hypothesis evaluation function.

Using the above results we can now determine the probability that k-overlap partition will be formed when categorizing a modal world. Let us compute the ratio between the total uncertainty the split (the k-overlap partition) and single category categorizations. Combining the above results and including the necessary normalization term yields the following expression for this ratio (referred to by ρ):

$$\rho = \left[1 + \left(\frac{\lambda}{1-\lambda}\right)\frac{1}{\log n}\right]\left[\log n - \frac{k}{n}\log k - \frac{n-k}{n}\log(n-k)\right] \qquad (5.7)$$

ρ represents the decision function used in category splitting step (6) of the categorization algorithm for the restricted case of two modal classes. If ρ is less than 1.0 then the uncertainty of the split categorization is less than the single category categorization and is thus to be accepted. Notice that ρ increases with λ. That is, it is more difficult to split a category when the value of λ is high. This behavior is to be expected. Higher values of λ cause the uncertainty measure U to weight the category uncertainty U_C more heavily than the property uncertainty U_P; coarse categorizations with few categories are preferred over finer categorizations. Thus a higher value of λ makes it more difficult to accept a split categorization over a single category.

Using equation 5.7 we can compute the maximum k such that a k-overlap partition has lower total uncertainty U than the single category categorization. Table 5.4 lists the maximum k for different values of n and λ; the fractional below is the proportion of equal size partitions of a set of $2n$ objects that are k-overlap partitions for k less than or equal to the maximum. For example, when there are 16 total objects ($n = 8$) and $\lambda = .4$, the maximum acceptable value of k is 2. Thus, k-overlaps of $k \in \{0, 1, 2\}$ have a lower uncertainty than the single category categorization. As indicated, this set of partitions constitutes 13% of the all partitions of $2n$ objects into equal-sized

n: / Lambda:	4	5	6	8	10	15	20
0.10	1 0.40571	1 0.20635	2 0.56710	3 0.61927	3 0.17090	6 0.46609	8 0.34307
0.20	1 0.40571	1 0.20635	1 0.00009	2 0.13193	3 0.17090	5 0.14311	7 0.11203
0.30	1 0.40571	1 0.20635	1 0.00009	2 0.13193	3 0.17090	4 0.02604	6 0.02564
0.40	0 0.02057	1 0.20635	1 0.00009	2 0.13193	2 0.02301	4 0.02604	5 0.00305
0.50	0 0.02057	0 0.00794	1 0.00009	1 0.01010	2 0.02301	3 0.00201	5 0.00305
0.60	0 0.02057	0 0.00794	0 0.00216	1 0.01010	1 0.00109	3 0.00201	4 0.00036
0.70	0 0.02057	0 0.00794	0 0.00216	1 0.01010	1 0.00109	2 0.00015	3 0.00002
0.80	0 0.02057	0 0.00794	0 0.00216	0 0.00016	0 0.00001	1 0.00000	2 0.00000
0.90	0 0.02857	0 0.00794	0 0.00216	0 0.00016	0 0.00001	0 0.00000	1 0.00000

Table 5.4: Maximum k such that the k-overlap partition has a lower uncertainty U than the single category categorization. The fractional number below represents the proportion of equal sized partitions of the $2n$ objects that are k-overlap partitions with k less than or equal to the maximum value.

categories. Notice that for $n \geq 15$ the percentage of acceptable partitions is almost zero for all but the lowest values of λ. Thus, we begin to see that category formation must occur before the initial category size ($2n$) becomes greater than 20 or so.

Using the maximum k values, we can compute the probability of success of the splitting step (6) of the categorization procedure. This probability depends upon how many partitions of the population are evaluated in each iteration of the algorithm; we let μ represent the number of attempts. We compute the *incremental probability* of success (incremental because it refers to only one iteration of the procedure) by computing the probability that

Incremental Success Probability Table

Hypotheses per iteration: 5

Lambda: n:	4	5	6	8	10	15	20
0.10	0.96402	0.68512	0.98480	0.99200	0.62676	0.95662	0.87765
0.20	0.96402	0.68512	0.34123	0.50710	0.62676	0.53802	0.45043
0.30	0.96402	0.68512	0.34123	0.50710	0.62676	0.12718	0.12181
0.40	0.13492	0.68512	0.34123	0.50710	0.10989	0.12718	0.01909
0.50	0.13492	0.03906	0.34123	0.04950	0.10989	0.01399	0.01909
0.60	0.13492	0.03906	0.01078	0.04950	0.00545	0.01399	0.00180
0.70	0.13492	0.03906	0.01078	0.04950	0.00545	0.00073	0.00010
0.80	0.13492	0.03906	0.01078	0.00078	0.00005	0.00001	0.00000
0.90	0.13492	0.03906	0.01078	0.00078	0.00005	0.00000	0.00000

Hypotheses per iteration: 10

Lambda: n:	4	5	6	8	10	15	20
0.10	0.99871	0.90085	0.99977	0.99994	0.86069	0.99812	0.98503
0.20	0.99871	0.90085	0.56602	0.75705	0.86069	0.78657	0.69797
0.30	0.99871	0.90085	0.56602	0.75705	0.86069	0.23818	0.22879
0.40	0.25164	0.90085	0.56602	0.75705	0.20771	0.23818	0.03782
0.50	0.25164	0.07659	0.56602	0.09654	0.20771	0.02779	0.03782
0.60	0.25164	0.07659	0.02144	0.09654	0.01088	0.02779	0.00359
0.70	0.25164	0.07659	0.02144	0.09654	0.01088	0.00145	0.00019
0.80	0.25164	0.07659	0.02144	0.00155	0.00011	0.00003	0.00001
0.90	0.25164	0.07659	0.02144	0.00155	0.00011	0.00000	0.00000

Hypotheses per iteration: 20

Lambda: n:	4	5	6	8	10	15	20
0.10	1.00000	0.99017	1.00000	1.00000	0.98059	1.00000	0.99978
0.20	1.00000	0.99017	0.81166	0.94097	0.98059	0.95445	0.90878
0.30	1.00000	0.99017	0.81166	0.94097	0.98059	0.41963	0.40523
0.40	0.43996	0.99017	0.81166	0.94097	0.37228	0.41963	0.07420
0.50	0.43996	0.14731	0.81166	0.18376	0.37228	0.05481	0.07420
0.60	0.43996	0.14731	0.04241	0.18376	0.02164	0.05481	0.00717
0.70	0.43996	0.14731	0.04241	0.18376	0.02164	0.00290	0.00039
0.80	0.43996	0.14731	0.04241	0.00310	0.00022	0.00006	0.00001
0.90	0.43996	0.14731	0.04241	0.00310	0.00022	0.00000	0.00000

Table 5.5: Probability of a successful split if 5, 10, or 20 attempts are considered. This probability is for a single iteration.

none of the attempted partitions is a k-overlap partition of a sufficiently low k. Assuming an independent sampling of partitions, the probability of failure is simply the proportion of partitions that do not satisfy maximum k-overlap condition multiplied μ times. The probability of success is then one minus this failure rate. Table 5.5 lists values of the incremental probability of a successful a split as a function of n, λ and μ. For example, if there are $n = 8$ objects in each class, if λ has been set to .5, and if $\mu = 10$ partitions are attempted, then the incremental probability of success is 0.0965. It is important to note that because the probability of failure is the repeated product of a number less than one — the percentage of partitions that overlap too many objects of the modal classes — the incremental probability of success can be made arbitrarily close to unity by increasing μ.

To determine whether a single category categorization will at any point be split into k-overlap partition of two categories, we compute the probability that every iteration through the algorithm fails to split the single category. Let us define $p_f(n, \lambda, \mu)$ to be the probability of *failing* to split a category in a given iteration; p_f is equal to one minus the probability of success. Then, assuming independence between iterations, the probability of never succeeding in splitting the single category categorization is simply the product of the probability of failing at each step. Because the number of objects increases by one with each iteration of the categorization procedure, but our equations are only defined for an even total number of objects ($2n$) we approximate this product as follows:

$$\text{Prob. of success } |_{n_0,\lambda,\mu} = 1 - \prod_{n_0}^{\infty} [p_f(n, \lambda, \mu) \cdot p_f(n+1, \lambda, \mu)] \qquad (5.8)$$

where n_0 is the initial n, equal to half the size of the initial category. It can be shown that this approximation is conservative in that it under-estimates the probability of success. Thus, we can finally compute the probability that the incremental hypothesis generation method will correctly categorize a modal world of two classes.

Equation 5.8 allows us to compute the probability of a successful categorization for given starting n_0, λ, and μ; the results are displayed in Table 5.6. Several observations should be made about about these results. First, when the starting n_0 is small, and when μ equals 10 or 20, the probability of

Success Probability Table

Hypotheses per iteration: 5

Start n: / Lambda:	4	5	6	8	10	15	20
0.10	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.96393
0.20	1.00000	1.00000	1.00000	1.00000	1.00000	0.99841	0.60276
0.30	1.00000	1.00000	1.00000	0.99986	0.99859	0.90314	0.17584
0.40	0.99937	0.99768	0.98883	0.95377	0.81115	0.49557	0.07944
0.50	0.89540	0.87417	0.80123	0.57333	0.47836	0.14445	0.02729
0.60	0.58400	0.49957	0.47356	0.24747	0.17659	0.03489	0.00250
0.70	0.33660	0.20197	0.16048	0.10193	0.01733	0.00220	0.00013
0.80	0.22838	0.07178	0.02353	0.00635	0.00511	0.00010	0.00000
0.90	0.22449	0.06709	0.01860	0.00134	0.00009	0.00000	0.00000

Hypotheses per iteration: 10

Start n: / Lambda:	4	5	6	8	10	15	20
0.10	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.99870
0.20	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.84220
0.30	1.00000	1.00000	1.00000	1.00000	1.00000	0.99062	0.32077
0.40	1.00000	0.99999	0.99988	0.99786	0.96434	0.74555	0.15257
0.50	0.98906	0.98417	0.96049	0.81795	0.72789	0.26803	0.05384
0.60	0.82695	0.74957	0.72286	0.43371	0.32200	0.06856	0.00499
0.70	0.55990	0.36314	0.29521	0.19346	0.03436	0.00439	0.00026
0.80	0.40460	0.13840	0.04650	0.01266	0.01020	0.00020	0.00001
0.90	0.39858	0.12969	0.03686	0.00267	0.00019	0.00000	0.00000

Hypotheses per iteration: 20

Start n: / Lambda:	4	5	6	8	10	15	20
0.10	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
0.20	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.97510
0.30	1.00000	1.00000	1.00000	1.00000	1.00000	0.99991	0.53864
0.40	1.00000	1.00000	1.00000	1.00000	0.99873	0.93526	0.28186
0.50	0.99988	0.99975	0.99844	0.96686	0.92596	0.46423	0.10479
0.60	0.97005	0.93729	0.92320	0.67931	0.54031	0.13242	0.00995
0.70	0.80632	0.59441	0.50327	0.34950	0.06754	0.00877	0.00052
0.80	0.64550	0.25765	0.09084	0.02515	0.02030	0.00040	0.00001
0.90	0.63829	0.24256	0.07235	0.00533	0.00037	0.00000	0.00000

Table 5.6: Probability of successful categorization of the two class modal world, for different starting values of n and different values of λ and μ.

success is quite high for most λ. (In the current implementation, $n_0 = 4$ and μ averages about 10.) Second, the success probability increases with μ, confirming our earlier statement that the probability of success can be raised by increasing μ. Finally, as n_0 becomes large, the probability of success drops rapidly. For example, for $\lambda = .6$ and $\mu = 10$, the probability of success drops from 32% to 7% as n_0 increases from 10 to 15.

To conclude our analysis, we consider two cases that deviate slightly from the previous conditions. First, is the case of more than two modal classes. Suppose there are 3 objects of each of four different classes (A, B, C, D) for a total of 12 objects. In the previous analysis, this situation corresponded to the case $n = 6$. Referring to table 5.4, we note that for $\lambda = .6$ only the 0-overlap partition is acceptable; there is only 1 such partition, for a probability of .002. In the new case of 4 classes, however, there are 3 possible partitions ($\{AB, CD\}$, $\{AC, BD\}$, $\{AD, BC\}$) that cause no overlap between classes. Each of these partitions yields a reduction in the property uncertainty with no corresponding increase in category uncertainty; each has a lower total uncertainty than the single category categorization. Thus, with more classes present the task of initially forming categories is easier.

The second variation is the case where the single category contains an *unequal* number of objects from the two classes; say n of one class of objects and m of another, where $n > m$. In this case the question arises of whether the observer attempts to form unequal sized partitions. Although doing so will permit him to possibly recover the exact partition, the increase in the possible number of partitions — the number of partitions is now on the order of 2^{n+m} as opposed to the previous case of 2^n — makes such a strategy unlikely to succeed. If, however, the observer only attempts equal sized partitions, then even the best possible partition will result in $n - m$ objects being in the wrong category. Thus, a smaller percentage of the partitions will be preferred over the single category than the case where there is an equal number of objects from each category. Recovering categories corresponding to the natural classes is more difficult when the objects are unevenly distributed in the initial categorization.

To summarize, we have determined the theoretical competence of the categorization procedure for the ideal case of a two-class, modal world. In particular, we have shown that the probability of successful categorization can be made arbitrarily high by increasing the number of partitions considered at each iteration (μ). Also, for values of μ used in the current implementation

and for a wide range of λ, the probability of success has been shown to be quite high. Finally, we have argued that if more than two classes of objects are present, the formation of categories is easier (for the same number of total objects) because a larger percentage of the possible partitions satisfy the k-overlap conditions.

5.4 Improving performance: Internal re-categorization

In the previous section we demonstrated that given a modal world, the probability of a successful categorization is high. However, often the probability is less than one. Also, real data is not purely modal, making categorization formation more difficult; noise features mask category structure. Thus we can expect errors of the form shown in Figure 5.2: two classes grouped together in one category.

The results of the previous section show that we can reduce the probability of this type of error by increasing μ — the number of category splits attempted during each iteration of the categorization procedure. However, the evaluation of a partition is an expensive computation.[15] Also, as n becomes large, the proportion of acceptable partitions is so small that we would require μ to be huge before a reasonable chance of success was attained. This sparse distribution of helpful partitions in the combinatoric space of possible partitions resulted in the poor performance of the random partitioning algorithm of Frotier and Solomon [1966]. Thus we would prefer a better solution.

One such possibility is simply re-categorizing each category in an incremental fashion. Consider again the last categorization shown in Figure 5.3. Let us assume that no split of any of the single class categories would be accepted, as is always true in the modal case. If we were to re-categorize each category independently, forming a new observation sequence for each, then

[15] In the current implementation the decision as to whether split a category is made locally. The program considers each category as its own population, and evaluates the proposed split relative to the single category categorization. The normalization factor used, however, is the one in based upon the total population. Otherwise, it would have the effect of scaling λ. This local decision can be shown to be approximately equivalent to deciding each split by considering the entire categorization of objects.

there is some (empirically shown to be high) probability that the poplar and the birch leaves would be properly separated. By repeated application of this procedure, the probability of a correct classification can again be made arbitrarily high.

Figures 5.6 and 5.7 illustrate an example of the implementation of such a re-categorization procedure. Figure 5.6 displays the result of executing the categorization procedure on a population of 80 leaves, 16 each of the species oak, maple, elm, birch, and poplar. The last categorization listed shows the resultant categorization formed after sequentially viewing the entire population.[16] In this example, the birch and the elm leaves have not been separated into distinct categories. However, we can re-execute the categorization procedure using the combined category to form a new observation sequence; when we do so, the categorization procedure correctly separates the classes (Figure 5.7).[17] Though not shown, we should mention that attempts to internally re-categorize the other categories do not yield any new divisions. The species categories approximate modal classes; any partition of the species produces a categorization with greater total uncertainty.

At this point we conclude our discussion of methods of improving the performance of the categorization procedure. There are two reason not to continue exploring methods of improving the statistical performance of the algorithm. First, the efficiency issues involved are not directly related to the question object categorization, but are more questions of statistical sampling; for example, the current implementation was improved by cycling through the objects sequentially, instead of generating an observation sequence by randomly sampling the population. Second, and more importantly, improving the behavior of the categorization procedure by increasing its efficiency does not address the fundamental question underlying object categorization: what information can be provided to the observer to facilitate the recovery of

[16]Notice that several oak leaves are missing. To make the implementation of the categorization procedure more robust, categories that become too small are deleted. Because every object is guaranteed to be repeated in the observation sequence, these deleted objects will be categorized again. Also, one birch leaf is contained in the poplar category. This particular object was mistakenly categorized early in the observation sequence. When it is viewed again, — the infinite observation sequence guarantees that it will be seen repeatedly — the mistake would be corrected.

[17]The normalization coefficient is *not* re-computed when re-categorizing the single category. Its purpose it to normalize U_P and U_C with respect to entire population.

Start:	MAPLE-23 POPLAR-51 BIRCH-69 BIRCH-72 POPLAR-60 BIRCH-70 BIRCH-67			
ELM-43:	BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60		ELM-43 MAPLE-23 BIRCH-70 BIRCH-72	
ELM-33:	BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60		ELM-33 ELM-43 MAPLE-23 BIRCH-70 BIRCH-72	
POPLAR-53:	POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60		ELM-33 ELM-43 MAPLE-23 BIRCH-70 BIRCH-72	
POPLAR-62:	POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60		ELM-33 ELM-43 MAPLE-23 BIRCH-70 BIRCH-72	
MAPLE-28:	POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60		MAPLE-28 ELM-33 ELM-43 MAPLE-23 BIRCH-70 BIRCH-72	
OAK-6:	POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60		OAK-6 MAPLE-28 ELM-33 ELM-43 MAPLE-23 BIRCH-70 BIRCH-72	
ELM-44:	POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	OAK-6 MAPLE-28 MAPLE-23 ELM-44	BIRCH-72 ELM-43 BIRCH-70 ELM-33	
OAK-13:	POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	OAK-13 OAK-6 MAPLE-28 MAPLE-23 ELM-44	BIRCH-72 ELM-43 BIRCH-70 ELM-33	

* * *

BIRCH-80:	POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	MAPLE-17 OAK-11 OAK-13 OAK-6 MAPLE-28 MAPLE-23 ELM-44	BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	
MAPLE-26:	POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	MAPLE-26 MAPLE-17 MAPLE-23 MAPLE-28	ELM-44 OAK-6 OAK-13 OAK-11
BIRCH-68:	POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	BIRCH-68 BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	MAPLE-26 MAPLE-17 MAPLE-23 MAPLE-28	ELM-44 OAK-6 OAK-13 OAK-11

* * *

OAK-16:	POPLAR-59 POPLAR-54 POPLAR-52 POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	ELM-46 ELM-34 ELM-40 BIRCH-68 BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	MAPLE-29 MAPLE-20 MAPLE-18 MAPLE-22 MAPLE-26 MAPLE-17 MAPLE-23 MAPLE-28	OAK-16 OAK-8 ELM-38 ELM-44 OAK-6 OAK-13 OAK-11
MAPLE-25:	POPLAR-59 POPLAR-54 POPLAR-52 POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	ELM-46 ELM-34 ELM-40 BIRCH-68 BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	MAPLE-25 MAPLE-29 MAPLE-20 MAPLE-18 MAPLE-22 MAPLE-26 MAPLE-17 MAPLE-23 MAPLE-28	OAK-16 OAK-8 ELM-38 ELM-44 OAK-6 OAK-13 OAK-11
ELM-47:	POPLAR-59 POPLAR-54 POPLAR-52 POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-67 BIRCH-69 POPLAR-51 POPLAR-60	ELM-47 ELM-46 ELM-34 ELM-40 BIRCH-68 BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	MAPLE-25 MAPLE-29 MAPLE-20 MAPLE-18 MAPLE-22 MAPLE-26 MAPLE-17 MAPLE-23 MAPLE-28	OAK-16 OAK-8 ELM-38 ELM-44 OAK-6 OAK-13 OAK-11

* * *

OAK-14:	POPLAR-63 POPLAR-61 POPLAR-56 POPLAR-55 POPLAR-50 POPLAR-57 POPLAR-58 POPLAR-49 POPLAR-59 POPLAR-54 POPLAR-52 POPLAR-64 POPLAR-62 POPLAR-53 BIRCH-69 POPLAR-51 POPLAR-60	ELM-41 BIRCH-76 BIRCH-77 ELM-39 BIRCH-71 ELM-42 ELM-45 ELM-37 ELM-35 BIRCH-79 BIRCH-74 ELM-48 BIRCH-66 BIRCH-78 BIRCH-65 BIRCH-73 ELM-47 ELM-46 ELM-34 ELM-40 BIRCH-68 BIRCH-80 BIRCH-75 ELM-36 BIRCH-72 ELM-43 BIRCH-70 ELM-33	MAPLE-32 MAPLE-27 MAPLE-30 MAPLE-31 MAPLE-19 MAPLE-21 MAPLE-24 MAPLE-25 MAPLE-29 MAPLE-20 MAPLE-18 MAPLE-22 MAPLE-26 MAPLE-17 MAPLE-23 MAPLE-28	OAK-14 OAK-5 OAK-7 OAK-2 OAK-10 OAK-3 OAK-15 OAK-12 OAK-9 OAK-1 OAK-16 OAK-11 OAK-4 OAK-6

Figure 5.6: Another example of the final categorization yielding a category containing two classes.

```
Resume:|BIRCH-75 BIRCH-69 BIRCH-70 ELM-35 BIRCH-74 ELM-38 ELM-45
BIRCH-73:|BIRCH-73 BIRCH-75 BIRCH-69 BIRCH-70 ELM-35 BIRCH-74 ELM-38 ELM-45
ELM-34:|ELM-34 BIRCH-73 BIRCH-75 BIRCH-69 BIRCH-70 ELM-35 BIRCH-74 ELM-38 ELM-45
ELM-36:|BIRCH-75 BIRCH-73 BIRCH-70 BIRCH-74 BIRCH-69        |ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
BIRCH-71:|BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70 BIRCH-74      |ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
         |BIRCH-69                                          |
ELM-33:|BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70 BIRCH-74        |ELM-33 ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
       |BIRCH-69                                            |
ELM-42:|BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70 BIRCH-74        |ELM-42 ELM-33 ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
       |BIRCH-69                                            |
BIRCH-67:|BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70      |ELM-42 ELM-33 ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
         |BIRCH-74 BIRCH-69                                 |
ELM-44:|BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70        |ELM-44 ELM-42 ELM-33 ELM-38 ELM-35 ELM-36 ELM-45
       |BIRCH-74 BIRCH-69                                   |ELM-34
ELM-41:|BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70        |ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35 ELM-36
       |BIRCH-74 BIRCH-69                                   |ELM-45 ELM-34
ELM-40:|BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70        |ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35
       |BIRCH-74 BIRCH-69                                   |ELM-36 ELM-45 ELM-34
BIRCH-80:|BIRCH-80 BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73      |ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35
         |BIRCH-70 BIRCH-74 BIRCH-69                        |ELM-36 ELM-45 ELM-34
BIRCH-72:|BIRCH-72 BIRCH-80 BIRCH-67 BIRCH-71 BIRCH-75      |ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35
         |BIRCH-73 BIRCH-70 BIRCH-74 BIRCH-69               |ELM-36 ELM-45 ELM-34
BIRCH-78:|BIRCH-78 BIRCH-72 BIRCH-80 BIRCH-67 BIRCH-71      |ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35
         |BIRCH-75 BIRCH-73 BIRCH-70 BIRCH-74 BIRCH-69      |ELM-36 ELM-45 ELM-34
BIRCH-79:|BIRCH-79 BIRCH-78 BIRCH-72 BIRCH-80 BIRCH-67      |ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35
         |BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70 BIRCH-74      |ELM-36 ELM-45 ELM-34
         |BIRCH-69                                          |
BIRCH-68:|BIRCH-68 BIRCH-79 BIRCH-78 BIRCH-72 BIRCH-80      |ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38 ELM-35
         |BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70      |ELM-36 ELM-45 ELM-34
         |BIRCH-74 BIRCH-69                                 |
ELM-39:|BIRCH-68 BIRCH-79 BIRCH-78 BIRCH-72 BIRCH-80        |ELM-39 ELM-40 ELM-41 ELM-44 ELM-42 ELM-33 ELM-38
       |BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70        |ELM-35 ELM-36 ELM-45 ELM-34
       |BIRCH-74 BIRCH-69                                   |
ELM-47:|BIRCH-68 BIRCH-79 BIRCH-78 BIRCH-72 BIRCH-80        |ELM-47 ELM-39 ELM-40 ELM-41 ELM-44 ELM-42 ELM-33
       |BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70        |ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
       |BIRCH-74 BIRCH-69                                   |
ELM-46:|BIRCH-68 BIRCH-79 BIRCH-78 BIRCH-72 BIRCH-80        |ELM-46 ELM-47 ELM-39 ELM-40 ELM-41 ELM-44 ELM-42
       |BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73 BIRCH-70        |ELM-33 ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
       |BIRCH-74 BIRCH-69                                   |
BIRCH-66:|BIRCH-66 BIRCH-68 BIRCH-79 BIRCH-78 BIRCH-72      |ELM-46 ELM-47 ELM-39 ELM-40 ELM-41 ELM-44 ELM-42
         |BIRCH-80 BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73      |ELM-33 ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
         |BIRCH-70 BIRCH-74 BIRCH-69                        |
ELM-37:|BIRCH-66 BIRCH-68 BIRCH-79 BIRCH-78 BIRCH-72        |ELM-37 ELM-46 ELM-47 ELM-39 ELM-40 ELM-41 ELM-44
       |BIRCH-80 BIRCH-67 BIRCH-71 BIRCH-75 BIRCH-73        |ELM-42 ELM-33 ELM-38 ELM-35 ELM-36 ELM-45 ELM-34
       |BIRCH-70 BIRCH-74 BIRCH-69                          |
```

Figure 5.7: Re-categorizing the single category containing the birch and elm leaves of Figure 5.6.

natural object categories. In the conclusion of this thesis, when we consider potential extensions to this work, we will return to the issue of recovering natural categories.

Chapter 6

Multiple Modes

The Principle of Natural Modes states that objects in the natural world cluster in dimensions important to the interaction between objects and their environment. However, clustering may occur at many levels: mammals and birds represent one natural grouping; cats and dogs another. This hierarchy of clusters — multiple modal levels — occurs because of hierarchical processes involved in the formation of objects; a "dog" may be described as a composite of the processes that create mammals and those that distinguish a dog from other mammals. Each process imposes a regularity on the objects, making inferences about the properties of these objects possible. For example, all mammals are warm-blooded and have hair.

We have claimed that the goal of the observer is to recover categories of objects corresponding to natural clusters. But this task is complicated by the presence of multiple modal levels. The properties constrained by one process may be independent of those constrained by another. Thus, if the different properties of objects encoded by the observer are constrained by different processes, then the category structure reflected in one set of properties is masked by other sets. The purpose of this chapter is to explore these issues.

We present a solution of the multiple mode problem that first entails identifying when multiple modes are present, and then incrementally segregating the population according to processes. We will continue to assume that objects are represented by property vectors and that the total categorization uncertainty U is used to measure the degree to which a categorization reflects the natural modes. U is defined as follows:

$$U(\mathcal{Z}) = (1 - \lambda)\, U_P(\mathcal{Z}) + \lambda\, \eta(\mathcal{Z})\, U_C(\mathcal{Z}) \tag{6.1}$$

where U_P is the uncertainty about the properties of an object once its category is known, U_C is the average uncertainty of the category to which an object belongs, η is a normalization coefficient between U_P and U_C, and λ is a free parameter representing the desired trade-off between the two uncertainties. (See chapter 4 for a complete definitions of these terms.) In this chapter we will consider the interaction between λ and the categories recovered by the observer. Also, the behavior of the incremental categorization algorithm presented in chapter 5 will be used to validate the theory developed here.

Our first step is to identify a null hypothesis, a case in which no modal structure is present. Being able to detect an unstructured situation will permit us to develop a strategy that relies on searching for sub-processes until no further structure can be found. Next, using both a simulation and a real example of a multiple mode population, we will examine the results of attempting to categorize such a set of objects; these results will suggest a method for separating modes according to processes. Finally, we will develop a method of evaluating the contribution of a feature to the recovery of a particular modal level.

6.1 A non-modal world

Natural modes are an appropriate basis for categorization because they represent classes of objects which are *redundant* in important properties. That is, from the knowledge of some properties of an object, the observer can infer a natural mode category that permits him to make inferences about other object properties. In an ideally modal world, the properties of interest to the observer — those he encodes about an object — are completely predictive: knowledge of one property permits predictions about all others.

In this section, however, we wish to define an *unstructured* world, a world with no natural modes. In such a world, the properties of objects are *independent*. Knowledge of some properties about an object provides no information about any of the other properties.[1] We refer to this world as a *non-modal world*. Our goal is to identify such non-modal worlds and to understand

[1] We are assuming that the properties encoded by the observer can be independent. A trivial counter example is when one property is the length of an object and another is

the behavior of the categorization algorithm and of the total uncertainty measure U when operating in such a world.

6.1.1 Categorizing a non-modal world: an example

We begin with a simulation of a non-modal world. We construct a population of 64 objects, described by six, independent, uniformly distributed, binary features. In this world an object is referred to as NULL-8, NULL-24, etc.; the property vector attached to each object is generated randomly from the the $2^6 = 64$ possibilities. These objects and features satisfy the non-modal condition in that knowledge about some of the properties of an object provides no information about any other properties. Next, we categorize these objects using an incremental categorization procedure that implements the total uncertainty measure U as a categorization evaluation function. (For a detailed discussion of the operation of the categorization algorithm see chapter 5.) The dynamic output of the categorization system is displayed in figures Figure 6.1 and Figure 6.2.

Figure 6.1 presents the results of executing the categorization algorithm with a value of λ of .55. Notice that categories continue to split into smaller categories as new objects are added; the last categorization shown contains only categories yet too small to be subdivided by the categorization algorithm. In the limit, the finest categorization — the categorization in which each object is its own category — would be selected. Because reducing the value of λ causes the categorization algorithm to produce only finer categorizations, we know that for all $\lambda \leq .55$, the finest categorization will be recovered. Figure 6.2 displays the results of running the categorization algorithm on the same population but with a λ of .6. Now, the stable categorization is one in which all objects are in a single category, the coarsest categorization possible. Reasoning as before, we know that for $\lambda \geq .6$ only the coarsest categorization will be recovered.[2] Thus, for this simulation of a

---

square of the length. A more subtle case is when one property is the area covered by an object, and another is the perimeter. (The perimeter must be greater than or equal to $2\sqrt{A\pi}$.) An interesting question is how the observer determines whether redundancy is caused by natural modes or logical dependence. A simple, though unexplored, solution relies on the fact that logical redundancies must be true for all objects, whereas modal dependencies hold only within the particular mode.

[2]For $.55 < \lambda < .6$ an intermediate categorization may be recovered, but it is unstable in

```
Start:|NULL-8 NULL-47 NULL-14 NULL-25 NULL-28 NULL-15 NULL-30
------------------------------------------------------------------------------------------------------------------------
NULL-39:|NULL-25 NULL-28 NULL-14 NULL-30                    |NULL-15 NULL-47 NULL-8 NULL-39
------------------------------------------------------------------------------------------------------------------------
 NULL-3:|NULL-25 NULL-28 NULL-14 NULL-30                    |NULL-3 NULL-15 NULL-47 NULL-8 NULL-39
------------------------------------------------------------------------------------------------------------------------
NULL-53:|NULL-25 NULL-28 NULL-14 NULL-30                    |NULL-53 NULL-3 NULL-15 NULL-47 NULL-8 NULL-39
------------------------------------------------------------------------------------------------------------------------
 NULL-4:|NULL-25 NULL-28 NULL-14 NULL-30                    |NULL-4 NULL-53 NULL-3 NULL-15 NULL-47 NULL-8
        |                                                  |NULL-39
------------------------------------------------------------------------------------------------------------------------
NULL-32:|NULL-32 NULL-25 NULL-28 NULL-14 NULL-30            |NULL-4 NULL-53 NULL-3 NULL-15 NULL-47 NULL-8
        |                                                  |NULL-39
------------------------------------------------------------------------------------------------------------------------
NULL-63:|NULL-32 NULL-25 NULL-28 NULL-14  |NULL-15 NULL-39 NULL-53 NULL-63  |NULL-47 NULL-3 NULL-4 NULL-8
        |NULL-30                          |                                 |

                                          *    *    *

------------------------------------------------------------------------------------------------------------------------
 NULL-1:|NULL-32 NULL-25 NULL-28 NULL-14  |NULL-29 NULL-57 NULL-59 NULL-15  |NULL-1 NULL-19 NULL-12 NULL-47
        |NULL-30                          |NULL-39 NULL-53 NULL-63          |NULL-3 NULL-4 NULL-8
------------------------------------------------------------------------------------------------------------------------
NULL-31:|NULL-32 NULL-25 NULL-28  |NULL-1 NULL-19 NULL-12   |NULL-31 NULL-15 NULL-39  |NULL-63 NULL-29 NULL-57
        |NULL-14 NULL-30          |NULL-47 NULL-3 NULL-4    |NULL-53                  |NULL-59
        |                         |NULL-8                   |                         |
------------------------------------------------------------------------------------------------------------------------
 NULL-7:|NULL-32 NULL-25    |NULL-31 NULL-15    |NULL-63 NULL-29    |NULL-4 NULL-12     |NULL-7 NULL-47
        |NULL-28 NULL-14    |NULL-39 NULL-53    |NULL-57 NULL-59    |NULL-8 NULL-19     |NULL-1 NULL-3
        |NULL-30            |                   |                   |                   |
------------------------------------------------------------------------------------------------------------------------
NULL-48:|NULL-32 NULL-25    |NULL-63 NULL-29    |NULL-4 NULL-12     |NULL-53 NULL-39    |NULL-15 NULL-3
        |NULL-28 NULL-14    |NULL-57 NULL-59    |NULL-8 NULL-19     |NULL-48 NULL-47    |NULL-1 NULL-7
        |NULL-30            |                   |                   |NULL-31            |

                                          *    *    *

------------------------------------------------------------------------------------------------------------------------
 NULL-5:|NULL-42      |NULL-61      |NULL-5       |NULL-62      |NULL-39      |NULL-4 NULL-8|NULL-36
        |NULL-32      |NULL-63      |NULL-15      |NULL-21      |NULL-47      |NULL-12      |NULL-51
        |NULL-25      |NULL-29      |NULL-3 NULL-1|NULL-31      |NULL-48      |NULL-16      |NULL-35
        |NULL-28      |NULL-57      |NULL-7       |NULL-23      |NULL-53      |             |NULL-52
        |NULL-14      |NULL-59      |             |             |             |             |NULL-19
        |NULL-30      |             |             |             |             |             |NULL-24

                                          *    *    *

------------------------------------------------------------------------------------------------------------------------
NULL-20:|NULL-6  |NULL-64  |NULL-4  |NULL-56  |NULL-60  |NULL-23  |NULL-30  |NULL-42  |NULL-33  |NULL-51
        |NULL-10 |NULL-39  |NULL-8  |NULL-40  |NULL-57  |NULL-21  |NULL-14  |NULL-58  |NULL-19  |NULL-20
        |NULL-9  |NULL-47  |NULL-12 |NULL-34  |NULL-62  |NULL-29  |NULL-32  |NULL-25  |NULL-27  |NULL-55
        |NULL-13 |NULL-48  |NULL-16 |NULL-50  |NULL-61  |NULL-17  |NULL-28  |NULL-44  |NULL-35  |NULL-24
        |NULL-5  |NULL-53  |        |NULL-52  |NULL-59  |NULL-31  |         |         |         |
        |NULL-15 |         |        |NULL-36  |NULL-63  |         |         |         |         |
        |NULL-3  |         |        |         |         |         |         |         |         |
        |NULL-1  |         |        |         |         |         |         |         |         |
        |NULL-7  |         |        |         |         |         |         |         |         |
```

Figure 6.1: The output of the categorization system when categorizing an independent world. For this execution the value of λ is .55. The categories generated continually split once they are large enough to be subdivided by the categorization algorithm. In the limit the preferred categorization consists of each object being its own category.

```
      Start:|NULL-28 NULL-52 NULL-34 NULL-54 NULL-9 NULL-17 NULL-13
    NULL-48:|NULL-34 NULL-54 NULL-52 NULL-48                      |NULL-28 NULL-13 NULL-9 NULL-17
     NULL-5:|NULL-34 NULL-54 NULL-52 NULL-48                      |NULL-5 NULL-28 NULL-13 NULL-9 NULL-17
    NULL-47:|NULL-34 NULL-54 NULL-52 NULL-48                      |NULL-47 NULL-5 NULL-28 NULL-13 NULL-9 NULL-17
    NULL-45:|NULL-34 NULL-54 NULL-52 NULL-48                      |NULL-45 NULL-47 NULL-5 NULL-28 NULL-13 NULL-9
            |                                                     |NULL-17
    NULL-59:|NULL-34 NULL-54 NULL-52 NULL-48  |NULL-59 NULL-13 NULL-45 NULL-47  |NULL-9 NULL-5 NULL-17 NULL-28
    NULL-11:|NULL-34 NULL-54 NULL-52 NULL-48  |NULL-11 NULL-59 NULL-13 NULL-45  |NULL-9 NULL-5 NULL-17 NULL-28
            |                                 |NULL-47                          |
                                          * * * *
    NULL-41:|NULL-42 NULL-38 NULL-34  |NULL-13 NULL-15 NULL-29  |NULL-27 NULL-47 NULL-59  |NULL-41 NULL-3 NULL-8
            |NULL-54 NULL-52 NULL-48  |NULL-45                  |NULL-11                  |NULL-39 NULL-19 NULL-23
            |                         |                         |                         |NULL-7 NULL-21 NULL-1
            |                         |                         |                         |NULL-6 NULL-5 NULL-17
            |                         |                         |                         |NULL-35 NULL-51 NULL-32
            |                         |                         |                         |NULL-28 NULL-9 NULL-18
            |                         |                         |                         |NULL-10
    NULL-31:|NULL-42 NULL-38 NULL-34 NULL-54  |NULL-41 NULL-3 NULL-8 NULL-39    |NULL-31 NULL-27 NULL-47 NULL-59
            |NULL-52 NULL-48                  |NULL-19 NULL-23 NULL-7 NULL-21   |NULL-11 NULL-13 NULL-15 NULL-29
            |                                 |NULL-1 NULL-6 NULL-5 NULL-17     |NULL-45
            |                                 |NULL-35 NULL-51 NULL-32 NULL-28  |
            |                                 |NULL-9 NULL-18 NULL-10           |
    NULL-37:|NULL-42 NULL-38 NULL-34 NULL-54 NULL-52 NULL-48    |NULL-37 NULL-41 NULL-3 NULL-8 NULL-39 NULL-19
            |                                                   |NULL-23 NULL-7 NULL-21 NULL-1 NULL-6 NULL-5 NULL-17
            |                                                   |NULL-35 NULL-51 NULL-32 NULL-28 NULL-9 NULL-18
            |                                                   |NULL-10 NULL-31 NULL-27 NULL-47 NULL-59 NULL-11
            |                                                   |NULL-13 NULL-15 NULL-29 NULL-45
    NULL-49:|NULL-42 NULL-38 NULL-34 NULL-54 NULL-52 NULL-48    |NULL-49 NULL-37 NULL-41 NULL-3 NULL-8 NULL-39
            |                                                   |NULL-19 NULL-23 NULL-7 NULL-21 NULL-1 NULL-6 NULL-5
            |                                                   |NULL-17 NULL-35 NULL-51 NULL-32 NULL-28 NULL-9
            |                                                   |NULL-18 NULL-10 NULL-31 NULL-27 NULL-47 NULL-59
            |                                                   |NULL-11 NULL-13 NULL-15 NULL-29 NULL-45
                                                  * * * *
    NULL-26:|NULL-36 NULL-42 NULL-38 NULL-34 NULL-54 NULL-52    |NULL-26 NULL-53 NULL-63 NULL-30 NULL-2 NULL-16
            |NULL-48                                            |NULL-49 NULL-37 NULL-41 NULL-3 NULL-8 NULL-39
            |                                                   |NULL-19 NULL-23 NULL-7 NULL-21 NULL-1 NULL-6 NULL-5
            |                                                   |NULL-17 NULL-35 NULL-51 NULL-32 NULL-28 NULL-9
            |                                                   |NULL-18 NULL-10 NULL-31 NULL-27 NULL-47 NULL-59
            |                                                   |NULL-11 NULL-13 NULL-15 NULL-29 NULL-45
    NULL-20:|NULL-20 NULL-26 NULL-53 NULL-63 NULL-30 NULL-2 NULL-16 NULL-49 NULL-37 NULL-41 NULL-3 NULL-8 NULL-39
            |NULL-19 NULL-23 NULL-7 NULL-21 NULL-1 NULL-6 NULL-5 NULL-17 NULL-35 NULL-51 NULL-32 NULL-28 NULL-9
            |NULL-18 NULL-10 NULL-31 NULL-27 NULL-47 NULL-59 NULL-11 NULL-13 NULL-15 NULL-29 NULL-45 NULL-36 NULL-42
            |NULL-38 NULL-34 NULL-54 NULL-52 NULL-48
    NULL-12:|NULL-12 NULL-20 NULL-26 NULL-53 NULL-63 NULL-30 NULL-2 NULL-16 NULL-49 NULL-37 NULL-41 NULL-3 NULL-8
            |NULL-39 NULL-19 NULL-23 NULL-7 NULL-21 NULL-1 NULL-6 NULL-5 NULL-17 NULL-35 NULL-51 NULL-32 NULL-28
            |NULL-9 NULL-18 NULL-10 NULL-31 NULL-27 NULL-47 NULL-59 NULL-11 NULL-13 NULL-15 NULL-29 NULL-45 NULL-36
            |NULL-42 NULL-38 NULL-34 NULL-54 NULL-52 NULL-48
```

Figure 6.2: The output of the categorization system when categorizing an independent world. For this execution the value of λ is .6. In this case the categorization produced consists of a single category.

non-modal world, no structured categorizations are recovered.

The fact that the categorization algorithm found no intermediate structure in the simulation of a non-modal world is encouraging: the algorithm should not impose structure on a world that contains none. A brief analysis of the non-modal world will demonstrate that in general the categorization algorithm will not recover structured categorizations when no natural clusters are present.

6.1.2 Theory of categorization in a non-modal world

Our goal is to explain the behavior of the categorization algorithm observed in the simulation of a non-modal world. To begin, let us consider the best categorizations one could construct in such a world. For example, what would be the best set of two categories one could form in a non-modal world described by binary features? The most structure one could impose would be to sort the population according to one feature. Because all the features are independent, members of a category would not have any other properties in common. Note that if there are m features, then there are m possible categorizations keeping one feature constant. Likewise, the best set of four categories would be one in which two features are held constant within each category; there are $\binom{m}{2}$ such possible categorizations. We refer to the number of features held constant in a categorization of a non-modal world as the *level* of the categorization. Figure 6.3 displays the different levels of categorizations for a non-modal world containing only 8 objects. Within each category is a property vector describing the members; an "x" indicates either a 1 or a 0. To indicate that there are several possible categorizations at each level, the feature held constant at level 1 is not held constant at level 2.

To understand the behavior of the categorization algorithm in such a non-modal world, let us consider how the two components of the evaluation function U — the property uncertainty U_P and the category uncertainty U_C — vary as we change categorization level. Let us expand our non-modal world to contain 128 objects described by 7 independent, uniformly distributed, binary features; we evaluate U_P and U_C for each possible categorization level 0 through 7. Panel (a) of Figure 6.4 displays the results of the evaluation; U_C

---

that it is not repeatable and a small change in λ will force the system to either of the two extreme categorizations.

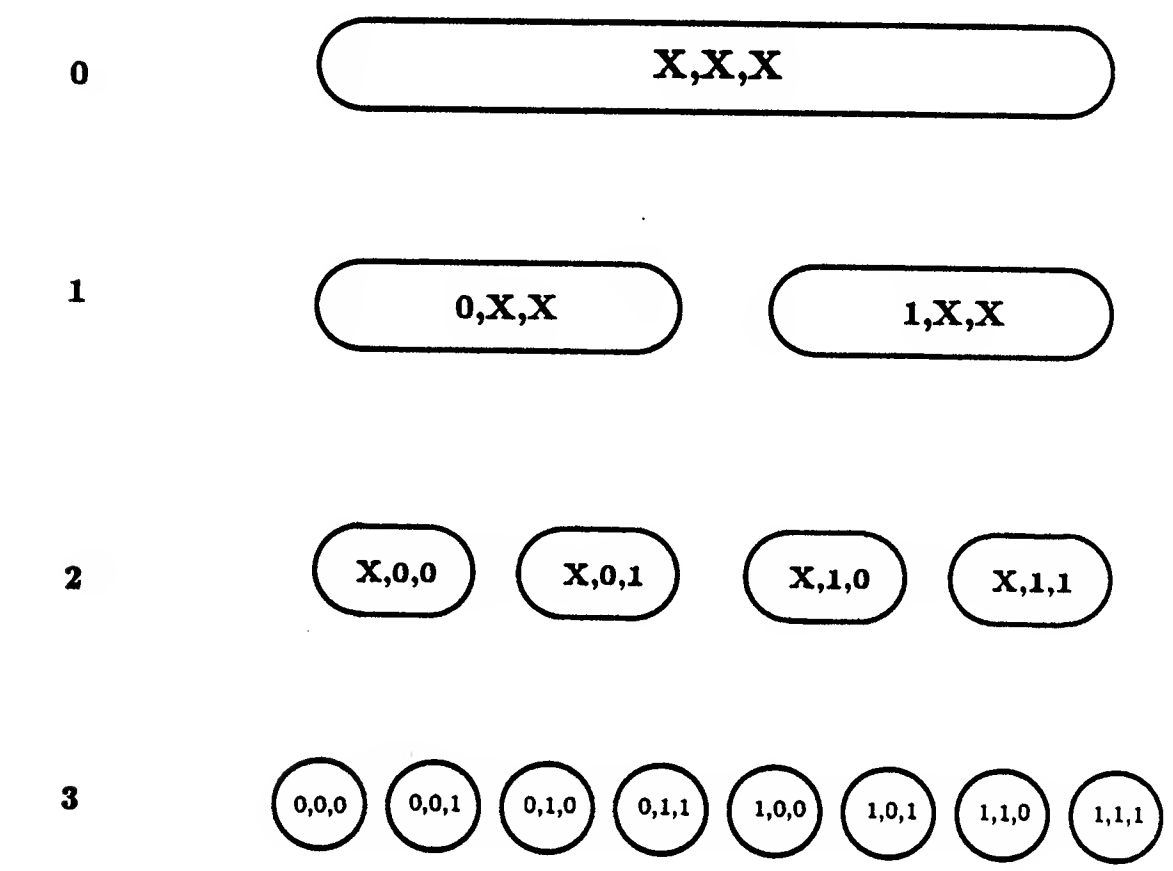
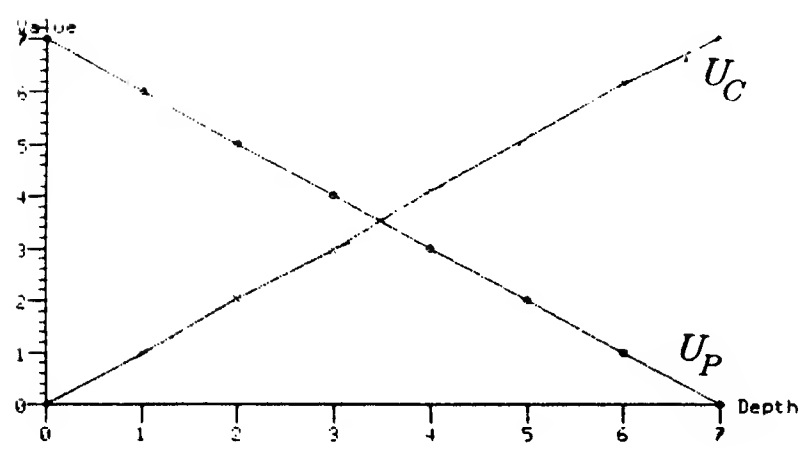
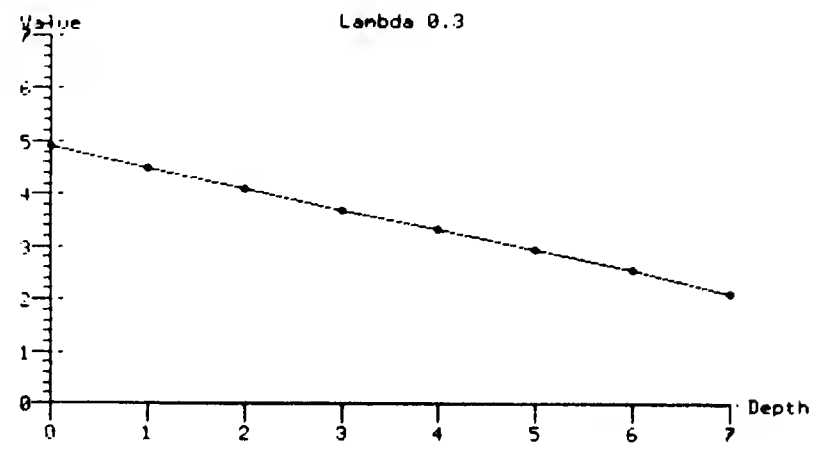


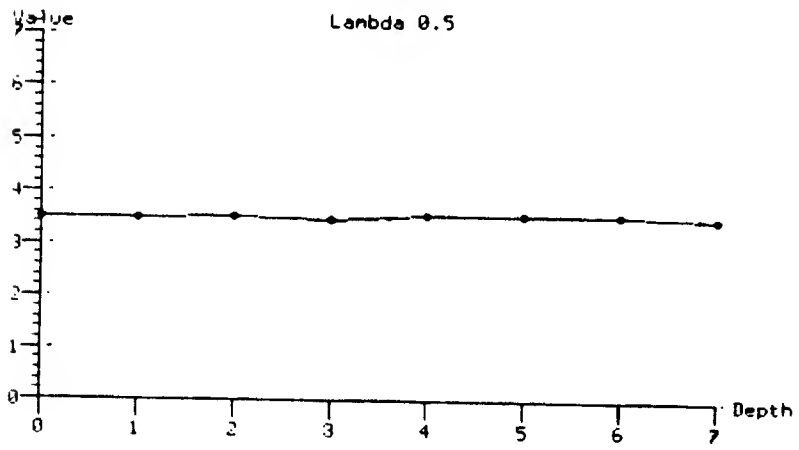
Figure 6.3: Different categorizations of a non-modal world of 8 objects, formed by varying the number of features held constant. Inside each category is a property vector describing the members; an "x" represents a feature that varies within the category.



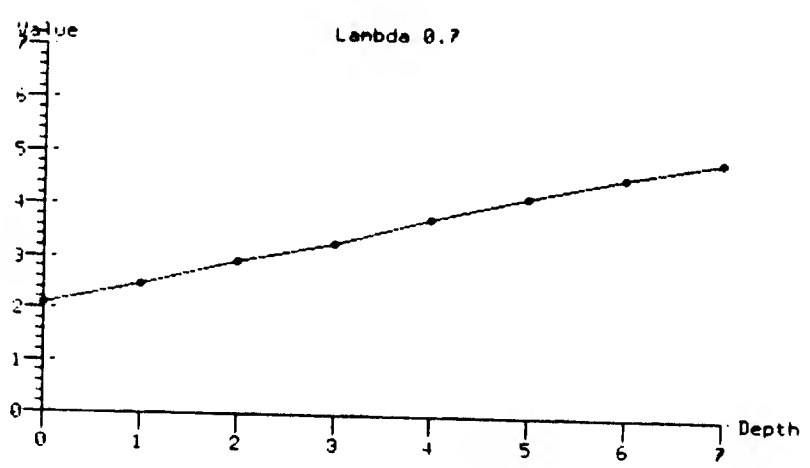
(a)



(b)



(c)



(d)

Figure 6.4: (a) The evaluation of U_P and (normalized) U_C for the different levels of categorization of a non-modal world consisting of 128 objects. The level is equal to the number of features held constant, which is also the logarithm of the number of categories. (b – d) The value of $U = (1 - \lambda)U_P + \eta\lambda U_C$ for λ of .3, .5, and .7.

has been normalized by η to be of the same scale as U_P. Panels (b–d) display the value of U for λ equal to .3, .5, and .7. Notice that as level increases — as more features are held constant in each category — the value of U_P decreases linearly while U_C increases, also linearly. Because both curves are linear and because they are normalized to be of the same scale, the change in U_P is equal to the negative of the change in U_C. As illustrated in panels b–d, the net change in U depends upon the value of λ. If λ is greater than .5, total uncertainty increases with level; for λ less than .5, total uncertainty decreases. The value of U is constant when λ equals .5.

The linearity of the graphs of U_P and U_C is predicted by analysis; these graphs were actually generated by simulating an independent world and evaluating different categorization levels. Each increase in categorization level is formed by keeping one more feature constant. Thus the property uncertainty is decreased by an amount reflecting the uncertainty of that feature. Because all the the features are binary and uniformly distributed, the decrease is $\log 2 = 1$. The initial value of U_P, at level 0, U_P is $7 \log 2 = 7$; the final value is zero. Similarly, with each increase in categorization level, the number of categories is doubled, and a greater number of properties is required to determine the category to which an object belongs. It can be shown that the increase in category uncertainty for each increase in categorization level is $\log \sqrt{2} = .5$. Thus, the value of η — the normalization factor between the two uncertainties — is 2. Let L be the the categorization level, and let L_{Max} be the maximum possible level; in our simulation with 128 objects $L_{Max} = 7$. Combining the above results for the two uncertainties and letting L be the level of categorization yields the following equation for the total uncertainty U:

$$U = (1-\lambda)(1.0)(L_{Max} - L) + (2.0)\lambda(.5)L = (2\lambda - 1)L + L_{Max}(1-\lambda) \quad (6.2)$$

That is, U is linear in L, and the slope is determined solely by λ. When $\lambda > .5$ the slope is positive; when $\lambda < .5$, the slope is negative.

The implication of the results in Figure 6.4 is that for an independent world the best categorization is either the coarsest partition, where all objects are in one category, or the finest, where each object is its own category. Which is preferable depends on the value of λ; for a value of .5 the decision is arbitrary.[3] Thus, we have explained the behavior of the categorization

[3]The simulation generated a critical λ greater than .5 because of an implementation mod-

algorithm displayed in the previous section. We have also demonstrated the competence of the incremental categorization algorithm in the non-modal world: the algorithm generates the correct solution.

The importance of the above result is that it allows the detection of the attempt to categorize a non-modal world. Because such a world contains no structure, there are no natural clusters, and no attempt at recovering modal categories should be made. In the next section we will make use of this diagnostic capability.

6.2 Multiple Modal Levels

6.2.1 Categorization stability

A ideal modal world is one in which all the properties are predictive of the natural classes and all the classes are predictive of the properties. The total uncertainty measure U has been constructed such that categorizations exhibiting a high degree of modal structure will produce low values of uncertainty. Because the incremental categorization algorithm developed in chapter 5 makes use of U as a categorization evaluation function, the algorithm recovers categories corresponding to modal clusters. To overcome the combinatoric problems in generating possible partitions of a population, the algorithm is stochastic, and thus is not guaranteed to find the correct modal solution.

However, the more modal a population — the closer the classes in the population approximate modal classes — the more reliable and repeatable is the categorization process; the modal categories will be recovered more often. This increase in reliability occurs because approximate categorizations, those that nearly separate the natural classes, are accepted during the incremental search for the best solution. Also, given an approximately modal world, a relatively wide range of λ will result, with high probability, in the incremental categorization algorithm discovering the modal categories. We use the term *categorization stability* to refer to the degree to which the recovery of a set

---

ification to the categorization algorithm. Specifically, merging occurs only if a merged categorization is sufficiently better than the split categorization, where "sufficiently" is determined by threshold. Using a non-zero threshold imparts some hysteresis to the system and overcomes instability caused by numerical inaccuracies.

	Length	Width	Flare	Lobes	Margin	Apex	Base	Color
Maple	3–6	3–5	0	5	Entire	Acute	Truncate	Light
Poplar	1–3	1,2	0,1	1	Crenate, Serrate	Acute	Rounded	Yellow
Oak	5–9	2–5	0	7,9	Entire	Rounded	Cumeate	Light
Elm	4–6	2,3	0,-1	1	Doubly Serrate	Accuminate	Rounded	Dark

Table 6.1: Leaf specifications for four species of leaves. A leaf generator creates property vectors consistent with the different specifications.

of categories is repeatable and insensitive to changes in λ.

As an example of a population exhibiting categorization stability we present the example of categorizing leaves. For this example, the property specifications for four species of leaves are generated according to descriptions provided by Preston [1976]. (Table 6.1) The properties chosen are known to be diagnostic of leaf species and thus are sufficient for the categorization task. (For an explanation of the details of object generation using property specifications see chapter 5.)

When categorizing a population of objects generated according to these specifications, the correct categories are reliably recovered for $.5 < \lambda < .75$.[4] Figure 6.5 displays the results for two such executions of the algorithm; for these examples λ was set to .6 and .55. In both cases the correct categories, those corresponding to the species, are recovered.

Outside this stable range, however, the correct categories are not recovered. Values of λ near one cause the uncertainty measure to prefer coarse categories, with high property uncertainty but low categorization uncertainty. In the case of the leaves, values of λ greater than .75 make it unlikely that the categorization algorithm will discover a partition into two categories that has a lower uncertainty than having the entire population in only one category. Thus the recovered categorization contains only one category. (For a

[4]An implementation detail: A λ of .75 is only successful when recovered categories are internally re-categorized, as discussed. The re-categorization, however, does not involve the re-computation of the normalization coefficient η. Doing so would have the effect of reducing λ because η decreases as a population is reduced, and its value directly multiplies the U_C term in U.

(a)

Start:	MAPLE-46 OAK-8 MAPLE-42 POPLAR-64 ELM-22 ELM-26 OAK-11		
MAPLE-39:	POPLAR-64 OAK-8 ELM-22 ELM-26		MAPLE-42 MAPLE-39 OAK-11 MAPLE-46
OAK-14:	POPLAR-64 OAK-8 ELM-22 ELM-26		OAK-14 MAPLE-42 MAPLE-39 OAK-11 MAPLE-46
POPLAR-61:	POPLAR-61 POPLAR-64 OAK-8 ELM-22 ELM-26		OAK-14 MAPLE-42 MAPLE-39 OAK-11 MAPLE-46
OAK-2:	POPLAR-61 POPLAR-64 OAK-8 ELM-22 ELM-26		OAK-2 OAK-14 MAPLE-42 MAPLE-39 OAK-11 MAPLE-46
MAPLE-43:	POPLAR-61 POPLAR-64 OAK-8 ELM-22 ELM-26		MAPLE-43 OAK-2 OAK-14 MAPLE-42 MAPLE-39 OAK-11 MAPLE-46
	* * *		
ELM-21:	MAPLE-44 MAPLE-36 MAPLE-40 MAPLE-41 MAPLE-34 MAPLE-38 MAPLE-48 MAPLE-47 MAPLE-33 MAPLE-45 MAPLE-35 MAPLE-37 MAPLE-46 MAPLE-43 MAPLE-39	OAK-10 OAK-13 OAK-12 OAK-4 OAK-3 OAK-5 OAK-7 OAK-16 OAK-6 OAK-15 OAK-9 OAK-1 OAK-11 OAK-2 OAK-14	POPLAR-54 POPLAR-56 POPLAR-55 POPLAR-62 POPLAR-58 POPLAR-50 POPLAR-51 POPLAR-60 POPLAR-57 POPLAR-63 POPLAR-49 POPLAR-59 POPLAR-64 POPLAR-53 POPLAR-61 OAK-8
			ELM-21 ELM-17 ELM-32 ELM-29 ELM-23 ELM-30 ELM-31 ELM-27 ELM-25 ELM-24 ELM-28 ELM-19 ELM-18 ELM-26 ELM-20 ELM-22 POPLAR-52

(b)

Start:	OAK-1 POPLAR-60 MAPLE-46 POPLAR-58 ELM-19 MAPLE-39 OAK-5		
MAPLE-37:	OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		OAK-5 ELM-19 POPLAR-60 POPLAR-58
MAPLE-45:	MAPLE-45 OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		OAK-5 ELM-19 POPLAR-60 POPLAR-58
OAK-2:	OAK-2 MAPLE-45 OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		OAK-5 ELM-19 POPLAR-60 POPLAR-58
POPLAR-62:	OAK-2 MAPLE-45 OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		POPLAR-62 OAK-5 ELM-19 POPLAR-60 POPLAR-58
POPLAR-49:	OAK-2 MAPLE-45 OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		POPLAR-49 POPLAR-62 OAK-5 ELM-19 POPLAR-60 POPLAR-58
OAK-4:	OAK-4 OAK-2 MAPLE-45 OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		POPLAR-49 POPLAR-62 OAK-5 ELM-19 POPLAR-60 POPLAR-58
POPLAR-53:	OAK-4 OAK-2 MAPLE-45 OAK-1 MAPLE-46 MAPLE-37 MAPLE-39		POPLAR-53 POPLAR-49 POPLAR-62 OAK-5 ELM-19 POPLAR-60 POPLAR-58
MAPLE-43:	POPLAR-53 POPLAR-49 POPLAR-62 OAK-5 ELM-19 POPLAR-60 POPLAR-58	MAPLE-45 MAPLE-37 MAPLE-46 MAPLE-43	OAK-1 OAK-2 OAK-4 MAPLE-39
ELM-18:	MAPLE-45 MAPLE-37 MAPLE-46 MAPLE-43	OAK-1 OAK-2 OAK-4 MAPLE-39	POPLAR-53 ELM-19 ELM-18 OAK-5
			POPLAR-49 POPLAR-62 POPLAR-60 POPLAR-58
	* * *		
POPLAR-56:	MAPLE-48 MAPLE-42 MAPLE-36 MAPLE-33 MAPLE-38 MAPLE-47 MAPLE-40 MAPLE-41 MAPLE-34 MAPLE-45 MAPLE-37 MAPLE-46 MAPLE-43	OAK-8 OAK-3 OAK-7 OAK-6 OAK-15 OAK-11 OAK-16 OAK-13 OAK-1 OAK-2 OAK-4	ELM-31 ELM-27 ELM-29 ELM-26 ELM-17 ELM-28 ELM-23 ELM-22 ELM-20 ELM-30 ELM-32 ELM-19 ELM-18 OAK-5
			POPLAR-56 POPLAR-63 POPLAR-57 POPLAR-61 POPLAR-50 POPLAR-55 POPLAR-49 POPLAR-62 POPLAR-60 POPLAR-58

Figure 6.5: Correctly categorizing four species of leaves. For these executions of the categorization algorithm the value of λ was set to .6 (a) and .55 (b). The identical results indicate the categorization is stable.

more complete analysis of the competence of the categorization algorithm, see Section 5.3.) Conversely, for λ below .5, the algorithm produces categories that are continually split, in the limit yielding the finest possible categorization. As shown in the previous section, this behavior is indicative of no useful internal structure; within each species, the property variations are independent.[5] This result is "correct" because the method that generates objects from property specifications does so by generating each property independently, and there is not structure present below the species level. This world of leaves has only one modal level of structure.

To summarize the leaves example, we have one region of λ that produces a stable categorization. Also, values of λ outside this range produce no useful category structure and the behavior of the categorization algorithm mimics the case when the world is independent. Recall that the setting of λ is established by the observer according to his goals. The value must be selected such that the categories provide a balance between the property uncertainty and categorization uncertainty which satisfies the inference requirements of the observer. If the λ of the observer lies within the stable range for the leaves world, then the correct categorization for him to recover is exactly the four species. If, however, the observer's particular value of λ lies outside this range then there exists no natural clustering of the objects that adequately supports his goals.

6.2.2 Multiple mode examples

We began this chapter by noting that the Principle of Natural Modes does not imply the existence of a unique clustering of objects. Rather, clusters which occur at different levels mirror the different levels of processes acting upon the objects. In this section we will explore the behavior of the categorization algorithm in the case of multiple levels of modal processes. First, we examine the results of attempting to categorize a real domain in which two levels of processes are apparent; these results will suggest that the properties constrained by the higher level process prevent the discovery of the lower level process. A simulation of an idealized two-process world will produce

[5]The behavior that is expected is the continual splitting of categories. The fact that it occurred around .5 is *not* significant and is purely a function of the data. The critical value of .5 derived in the previous section only applies when the entire population is independent.

	BF *bacteroides fragilis*	BT *bacteroides thetaiotamicron*	BV *bacteroides vulgatus*	FM *fusobacterium mortiferum*	FN *fusobacterium necrophorum*	FV *fusobacterium varium*
loc	GI	GI	GI	OR	OR	OR
gram	neg	neg	neg	neg	neg	neg
gr-pen	R	R	R	{R,S}	S	{R,S}
gr-rif	S	S	S	R	S	R
gr-kan	R	R	R	S	S	S
dole	neg	pos	neg	neg	pos	pos
esculin	pos	pos	neg	pos	neg	neg
bile	E	E	E	E	I	E
glc	ls	ls	ls	none	none	none
rham	neg	pos	pos	{neg,pos}	{neg,pos}	{neg,pos}
nf1	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}
nf2	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}

Table 6.2: The property specifications for six different species of bacteria. Because most of the real features have only one value per species, two noise features are added (**nf1** and **nf2**).

similar behavior in the categorization algorithm, confirming the masking of the lower level modes by the higher level properties.

The domain of infectious bacteria will serve to illustrate the behavior of the categorization algorithm in a two-process world. Property specifications for six different species of bacteria are encoded according to data taken from [Dowell and Allen, 1981]; Table 6.2 displays the specifications for these species. Because most of the real features take on only one value per species (unlike the leaves where features like "length" and "width" varied greatly) two noise features are added (**nf1** and **nf2**). These features prevent all objects of the same class from having identical property descriptions.

Of the six species, three are from the genus *bacteroides*; these are abbreviated as *BF*, *BT*, and *BV*. The other three — *FM*, *FN*, and *FV* — are from the genus *fusobacterium*. Notice that several of the features of the specifications are determined by the genus, while others are determined by the species. For example, all members of *bacteroides* have the property "gr-kan = R" (coding for "growth in presence of Kanamycin is resistant"). Other properties, such as "dole," vary between the species, ignoring genus boundaries. Still others, such as "gr-rif," are confounded between levels. These property

specifications are used to generate 12 instances of each type of bacteria.

Figure 6.6 displays the results of categorizing this population with values of λ of .65 and .7. Notice that in both cases the bacteria have been categorized according to their genus. This categorization is stable: repeated application of the algorithm consistently discovers the two genera. Furthermore, for $.6 < \lambda < .75$ the recovered categories correspond to the genera. If the λ of the observer falls within this range, then the natural categories corresponding to the genera satisfy the inference requirements of the observer.

Suppose however, the value of λ does not fall within this range. Figure 6.7 displays the results of of two executions of the categorization algorithm with a λ of 0.5. In this case the recovered categories do not correspond to either the genera or the species, but to a composite of the two levels. For example, in the first execution, (Figure 6.7a) the category on the left corresponds to a single species. The next category to the right is comprised of two species of the same genus. Finally, the last two categories are mixtures of two or more species, although they only contain instances of one genus. Also, if the categorization algorithm is repeated, different composite categories are recovered: the categorization is unstable. Furthermore, if the value of λ is decreased (say to 0.4) the categorization algorithm produces the finest possible categorization, indicating no modal structure for λ less than 0.5. That is, decreasing λ will *not* permit the categorization algorithm to recover the species.[6]

Compare this example with the leaves example of the previous section. When categorizing the leaves with a λ outside the stable range, the categorization algorithm produces either the coarsest or the finest categorization. This behavior is similar to that observed when the features in the world are independent and no natural classes exist. Thus we concluded that there existed only one modal level in that population. In the case of the bacteria, however, a λ outside the stable range produces (unstable) categories that are composites of the classes. Thus, the existence of these unstable categorizations indicates that competing structures — different modal levels — may be present in the population.

[6]We should note that internal re-categorization — without re-normalization — is not sufficient to resolve the problem. The difficulty in recovering the species lies with the masking of the species structure by the genus constrained properties, not in the statistical failure of the categorization algorithm. In the next section we will consider the case of internal re-categorization with re-normalization.

(a)

```
Start:|FV-61 BF-12 BF-9 FN-60 BV-26 BT-15 FM-38
BT-22:|FN-60 BT-15 FM-38 FV-61                                  |BF-12 BT-22 BF-9 BV-26
FN-54:|FN-54 FN-60 BT-15 FM-38 FV-61                            |BF-12 BT-22 BF-9 BV-26
BV-36:|FN-54 FN-60 BT-15 FM-38 FV-61                            |BV-36 BF-12 BT-22 BF-9 BV-26
FV-65:|FV-65 FN-54 FN-60 BT-15 FM-38 FV-61                      |BV-36 BF-12 BT-22 BF-9 BV-26
BV-29:|FV-65 FN-54 FN-60 BT-15 FM-38 FV-61                      |BV-29 BV-36 BF-12 BT-22 BF-9 BV-26
FM-41:|FM-41 FV-65 FN-54 FN-60 BT-15 FM-38 FV-61                |BV-29 BV-36 BF-12 BT-22 BF-9 BV-26
FM-37:|FM-37 FM-41 FV-65 FN-54 FN-60 BT-15 FM-38 FV-61          |BV-29 BV-36 BF-12 BT-22 BF-9 BV-26
BV-31:|FM-37 FM-41 FV-65 FN-54 FN-60 BT-15 FM-38 FV-61          |BV-31 BV-29 BV-36 BF-12 BT-22 BF-9 BV-26
FM-43:|BV-31 BV-29 BV-36 BF-12 BT-22     |FM-43 FM-41 FV-61 FM-37          |FM-38 FV-65 FN-60 FN-54 BT-15
      |BF-9 BV-26                         |                                 |
                                          * * * *
BV-25:|BV-25 BV-34 BF-7 BT-24 BT-21 BV-30 BT-13 BF-11      |FV-62 FN-50 FV-70 FV-69 FV-64 FM-44 FN-56 FM-48
      |BV-32 BT-23 BV-33 BF-5 BF-10 BF-3 BV-31 BV-29 BV-36|FM-47 FM-42 FV-68 FN-51 FN-52 FN-59 FV-72 FM-40
      |BF-12 BT-22 BF-9 BV-26                             |FM-38 FV-65 FN-60 FN-54 BT-15 FV-66 FM-43 FM-41
      |                                                   |FV-61 FM-37
FN-53:|BV-25 BV-34 BF-7 BT-24 BT-21 BV-30 BT-13 BF-11      |FN-53 FV-62 FN-50 FV-70 FV-69 FV-64 FM-44 FN-56
      |BV-32 BT-23 BV-33 BF-5 BF-10 BF-3 BV-31 BV-29 BV-36|FM-48 FM-47 FM-42 FV-68 FN-51 FN-52 FN-59 FV-72
      |BF-12 BT-22 BF-9 BV-26                             |FM-40 FM-38 FV-65 FN-60 FN-54 BT-15 FV-66 FM-43
      |                                                   |FM-41 FV-61 FM-37
```

(b)

```
Start:|BT-24 FM-39 BV-32 FV-68 BF-1 FM-47 FV-63
BV-27:|BV-32 FV-63 BT-24 BV-27                                  |FV-68 BF-1 FM-47 FM-39
FN-52:|BV-32 FV-63 BT-24 BV-27                                  |FN-52 FV-68 BF-1 FM-47 FM-39
BT-21:|BT-21 BV-32 FV-63 BT-24 BV-27                            |FN-52 FV-68 BF-1 FM-47 FM-39
FM-41:|BT-21 BV-32 FV-63 BT-24 BV-27                            |FM-41 FN-52 FV-68 BF-1 FM-47 FM-39
FM-40:|BT-21 BV-32 FV-63 BT-24 BV-27                            |FM-40 FM-41 FN-52 FV-68 BF-1 FM-47 FM-39
BV-28:|BV-28 BT-21 BV-32 FV-63 BT-24 BV-27                      |FM-40 FM-41 FN-52 FV-68 BF-1 FM-47 FM-39
BT-19:|BT-19 BV-28 BT-21 BV-32 FV-63 BT-24 BV-27                |FM-40 FM-41 FN-52 FV-68 BF-1 FM-47 FM-39
FN-50:|BT-19 BV-28 BT-21 BV-32 FV-63     |BF-1 FM-39 FM-47 FM-41          |FN-52 FV-68 FN-50 FM-40
      |BT-24 BV-27                        |                                |
                                          * * * *
BT-21:|BT-21 BV-33 BV-26 BF-2 BT-23 BV-31 BF-8 BV-32 BF-7 |FM-47 FV-62 FN-56 FM-41 FM-39 FN-52 FV-71 FV-67
      |BV-29 BF-12 BV-31 BF-10 BV-34 BF-5 BT-17 BT-13      |FV-65 FN-55 FN-49 FN-51 FV-69 FM-48 FM-44 FN-57
      |BV-35 BV-33 BT-20 BF-6 BF-7 BT-14 BT-23 BF-9 BT-15 |FN-55 FN-51 FN-49 FV-66 FN-54 FV-61 FN-60 FV-62
      |BT-18 BT-21 BF-8 BF-2 BT-19 BV-26 BV-36 BV-25 BV-27|FV-69 FN-52 FN-58 FN-50 FV-68 FM-40 FV-71 FN-53
      |BV-28 BV-30 BV-32 BT-24 BT-22 BF-3                  |FN-59 FM-38 FM-37 FM-46 FM-44 FM-41 FM-45 FV-72
      |                                                    |FM-39 FV-70 FN-56 FM-43 FM-38 FM-47 BF-1 FV-67
      |                                                    |FV-65 BF-4 FV-64 FV-63
```

Figure 6.6: Categorizing bacteria. In these examples λ equals (a) .65 and (b) .7. The categories recovered correspond to the different genera.

(a)

```
Start:|BV-29 FV-64 BV-25 FM-42 BV-30 BF-2 FV-68
BV-26:|BV-25 BV-30 BV-26 BV-29                                  |FV-68 FM-42 BF-2 FV-64
FM-46:|BV-25 BV-30 BV-26 BV-29                                  |FM-46 FV-68 FM-42 BF-2 FV-64
BT-16:|BT-16 BV-25 BV-30 BV-26 BV-29                            |FM-46 FV-68 FM-42 BF-2 FV-64
FV-70:|BT-16 BV-25 BV-30 BV-26 BV-29                            |FV-70 FM-46 FV-68 FM-42 BF-2 FV-64
BT-23:|BT-23 BT-16 BV-25 BV-30 BV-26 BV-29                      |FV-70 FM-46 FV-68 FM-42 BF-2 FV-64

                                * * * *

FM-40:|BV-33 BV-34 BV-35 BV-31  |BV-32 BT-19 BV-28 BT-22  |FM-37 FN-59 FM-48 FV-61  |FM-40 FM-45 FM-44 FM-46
      |BV-27 BV-25 BV-30 BV-36  |BT-20 BF-11 BF-12 BT-21  |FN-51 FV-71 FV-69 FN-52  |FV-66 FV-70 FV-72 FM-43
      |BV-26                    |BT-13 BF-8 BF-6 BF-7 BF-3|FN-54 FN-49 FN-55 FV-67  |FM-41 FM-39 FM-38 FM-42
      |                         |BT-24 BF-10 BT-14 BF-5   |FN-56 FM-47 FV-64 FN-58  |
      |                         |BF-1 BF-4 BT-18 BT-23    |FN-53 FV-65 FN-57 FN-60  |
      |                         |BT-16 BT-15 BT-17        |FV-68 FV-63 FN-50        |
```

(b)

```
Start:|BT-22 BV-27 FN-57 BT-18 FN-55 FN-60 BF-3
BV-30:|BV-30 FN-57 FN-60 FN-55                                  |BT-18 BF-3 BT-22 BV-27
FM-41:|FM-41 BV-30 FN-57 FN-60 FN-55                            |BT-18 BF-3 BT-22 BV-27
 BF-6:|FM-41 BV-30 FN-57 FN-60 FN-55                            |BF-6 BT-18 BF-3 BT-22 BV-27
BT-20:|FM-41 BV-30 FN-57 FN-60 FN-55                            |BT-20 BF-6 BT-18 BF-3 BT-22 BV-27
BF-12:|FM-41 BV-30 FN-57 FN-60 FN-55                            |BF-12 BT-20 BF-6 BT-18 BF-3 BT-22 BV-27

                                * * * *

BT-16:|BT-16 BV-36  |FN-51 FN-56  |FV-61 FV-70  |FN-54 FV-63  |FM-40 FM-38  |FV-69 FV-62  |FV-71 FV-67
      |BF-10 BV-26  |FM-55 FN-57  |FV-72 FN-52  |FV-65 FN-53  |FM-37 FM-39  |BV-30 FN-59  |FM-45 FV-64
      |BF-7 BV-31   |FN-60        |FN-58 FN-50  |FV-68        |FM-44 FM-46  |             |
      |BV-32 BT-19  |             |             |             |FM-47 FM-43  |             |
      |BT-13 BV-28  |             |             |             |             |             |
      |BV-34 BV-25  |             |             |             |             |             |
      |BT-24 BF-11  |             |             |             |             |             |
      |BT-14 BT-21  |             |             |             |             |             |
      |BT-23 BF-4   |             |             |             |             |             |
      |BV-33 BT-17  |             |             |             |             |             |
      |BF-2 BF-9    |             |             |             |             |             |
      |BF-1 BV-35   |             |             |             |             |             |
      |BF-5 BV-29   |             |             |             |             |             |
      |BF-8 BT-15   |             |             |             |             |             |
      |BF-12 BT-20  |             |             |             |             |             |
      |BF-6 BT-18   |             |             |             |             |             |
      |BF-3 BT-22   |             |             |             |             |             |
      |BV-27        |             |             |             |             |             |
```

Figure 6.7: Unstable categorization of bacteria. In these examples λ equals 0.5. The categories recovered do not correspond to either the genera or the species. For example, in (a) the left most category corresponds to all instances of of one species, but the other three categories are mixtures of species from one genus. This categorization is unstable; repeating the categorization procedure (b) yields a different composite.

	GS11 *genus-1 species-1-1*	GS12 *genus-1 species-1-2*	GS13 *genus-1 species-1-3*	GS21 *genus-2 species-2-1*	GS22 *genus-2 species-2-2*	GS23 *genus-2 species-2-3*
gf1	1	1	1	2	2	2
gf2	1	1	1	2	2	2
gf3	1	1	1	2	2	2
gf4	1	1	1	2	2	2
gf5	1	1	1	2	2	2
sf1	1	2	3	1	2	3
sf2	1	2	3	1	2	3
nf1	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}
nf2	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}	{1,2,3,4}

Table 6.3: The property specifications for a simulated two process world. Five features are modal for the genera (**gf1** – **gf5**), two are modal for the species (**sf1** and **sf2**), and two are noise (**nf1** and **nf2**).

To test the validity of this diagnostic inference, we simulate an ideal "two-process modal" world. In such a world, some of the features used to describe the objects are constrained by one modal "process," while others are constrained by a second modal process. Also, some noise features are included. Table 6.3 lists the property specifications for such a world. By "two-process modal" we mean that each feature is modal for the level at which it operates. For example, **gf1** (for "genus-feature-1") is modal for the different genera; it takes on a different value for each of the two genera. Likewise, **sf1** is modal *for each species within the genus.* For this particular example there are five features constrained by the genus and two by the species;[7] two noise features are also included. The importance of this simulation is that we know no features confound the two processes. In the bacteria example certain features (such as "gr-rif") appeared to be affected by both genus and species. If the categorization algorithm produces unstable categorizations in this simulation we know that such confounding properties are not required to produce this behavior and that unstable categorizations are an indicator of competing modal levels.

[7] In the next section we will explain why we created an imbalance between the number of features constrained by the genus and the number constrained by the species.

(a)

Start:	GS13-29	GS13-22	GS12-12	GS23-59	GS11-3	GS12-19	GS22-49												
GS23-56:	GS22-49	GS23-59	GS23-56	GS13-22				GS12-12	GS13-29	GS12-19	GS11-3								
GS23-53:	GS23-53	GS22-49	GS23-59	GS23-56	GS13-22			GS12-12	GS13-29	GS12-19	GS11-3								
GS13-26:	GS13-26	GS23-53	GS22-49	GS23-59	GS23-56	GS13-22													
GS11-10:	GS11-10	GS13-26	GS23-53	GS22-49	GS23-59	GS23-56	GS13-22												
GS11-2:	GS13-22	GS11-2	GS13-26	GS22-49				GS11-10	GS23-59	GS23-53	GS23-56								
GS22-48:	GS13-22	GS11-2	GS13-26	GS22-49				GS22-48	GS11-10	GS23-59	GS23-53	GS23-56							

GS13-28:	GS21-37	GS21-36		GS11-7	GS11-1	GS11-8		GS23-55	GS22-44			GS13-28	GS12-16						
	GS21-34	GS21-38		GS11-2	GS11-9	GS11-5		GS21-40	GS22-45			GS13-21	GS13-25						
	GS21-32	GS21-35		GS11-6				GS22-46	GS13-30			GS12-17	GS12-15						
	GS22-42	GS21-33						GS22-50	GS23-58			GS12-13	GS12-14						
	GS21-31							GS22-47	GS13-26			GS12-20	GS11-4	GS13-23					
								GS22-49	GS13-24			GS12-11	GS13-27						
								GS21-39	GS22-43			GS12-18	GS13-22						
								GS22-41	GS23-57										
								GS23-52	GS23-60										
								GS23-54	GS23-51										
								GS22-48	GS23-59										
								GS23-53	GS23-56										

(b)

Start:	GS12-13	GS13-21	GS21-35	GS22-49	GS21-34	GS13-25	GS13-23												
GS13-28:	GS22-49	GS13-21	GS21-34	GS21-35				GS13-25	GS13-23	GS13-28	GS12-13								
GS21-38:	GS21-38	GS22-49	GS13-21	GS21-34	GS21-35			GS13-25	GS13-23	GS13-28	GS12-13								
GS12-15:	GS21-38	GS22-49	GS13-21	GS21-34	GS21-35			GS12-15	GS13-25	GS13-23	GS13-28	GS12-13							
GS12-18:	GS21-38	GS22-49	GS13-21	GS21-34	GS21-35			GS12-18	GS12-15	GS13-25	GS13-23	GS13-28							
								GS12-13											
GS12-19:	GS21-38	GS22-49	GS13-21	GS21-34	GS21-35			GS12-19	GS12-18	GS12-15	GS13-25	GS13-23							
					* * * *			GS13-28	GS12-13										

GS21-33:	GS22-50	GS22-41		GS21-33	GS23-54			GS12-11	GS12-19			GS13-26	GS11-3	GS11-7					
	GS22-44	GS22-43		GS23-59	GS22-46			GS12-14	GS12-18			GS13-24	GS13-30	GS11-4					
	GS22-47	GS22-48		GS21-31	GS23-55			GS12-17				GS11-10	GS11-6	GS12-12					
	GS22-49			GS23-51	GS21-32							GS12-15	GS11-5	GS11-9					
				GS23-52	GS21-37							GS12-16	GS11-1	GS11-8					
				GS21-35	GS21-34							GS12-20	GS13-29						
				GS13-21	GS21-38							GS13-27	GS13-22						
				GS23-57	GS23-60							GS13-25	GS12-13						
				GS21-39	GS23-53							GS13-23	GS13-28						
				GS21-36	GS21-40														
				GS22-42	GS23-56														
				GS22-45															

Figure 6.8: The categorization of the simulated two-process modal world yields unstable categories when λ equals .5 The genus of each object is indicated by the first digit in the name; the species, by the second.

A population of 60 objects, 10 per species, is generated according to the specifications of Table 6.3. The population is then categorized for a wide range of λ. For λ greater than .8, the categorization algorithm produces only single-category categorizations. For $.7 \leq \lambda \leq .8$ a categorization corresponding to the simulated genera is recovered. This categorization was highly stable in that it was recovered on almost all categorization attempts for λ within this range. However, for $.5 \leq \lambda \leq .6$ unstable categorizations are formed; the categories recovered are approximately the union of some of the different species. Figure 6.8 displays the result of two executions of the categorization algorithm with λ equal to .5. The algorithm was interrupted after each of the objects had been viewed once. In Figure 6.8a, the four resulting categories closely correspond to the those that would be formed by combining the GS21 species with the GS23 species (both are of the same genus) as well as combining GS11 with GS13.[8] In the other categorization attempt (Figure 6.8b) a different set of categories is generated. Finally, for $\lambda \leq .4$ the algorithm produces the finest possible categorization, yielding no structured categories (not shown).

The results of this simulation support the conclusion that multiple modal levels yield unstable categorization behavior. Unfortunately, we cannot invoke the power of evolution to prevent this situation from arising. The observer must be able to recover the natural categories corresponding to a sufficiently structured process level to support necessary inferences. For example, if the differences in the species of bacteria are important to the observer then he will need to encode properties that discriminate between genera as well as properties that can distinguish the species within the genera. Thus, the observer requires a method of recovering the natural categories at *each* process level in the hierarchy until the appropriate level is achieved. In the next sections we will describe a possible method for recovering the natural categories at each level, and for determining the processes associated with each property encoded by the observer.

[8] If the algorithm were permitted to continue, subsequent viewing of the objects would help correct prior mistakes.

6.3 Process separation

6.3.1 Recursive categorization

Let us continue the simulation example of the previous section. If the goals of the observer require that he recover more than just the genera, then he needs a method by which to separate the species. We know that reducing the value of λ below the stable value that recovered the genera will not cause the categorization algorithm to reliably recover the species; the interaction between the species and genera cause composite categories to be formed.

Suppose, however, we consider each genus separately, as its own world of objects. In that case, the world contains only three classes of objects, namely the three species. Unlike the entire population, which contained more than one modal level, there is only one level of structure present in this world. If the observer can recover the modal structure within this population, then, by applying the procedure to both genera, he would be able to recover the categories corresponding to the species.

In this world there are three types of features: *modal*, where each feature takes on a different value for each class; *noise*, where the value is independent of the class; and *constant*, where the value never varies. The only differences between this world and previous examples in which there was only one modal level are the constant features. For example, the leaves domain contained several highly diagnostic (almost modal) features (such as "apex" and "base") as well as mostly unconstrained features (such as "length"). We know that the categorization algorithm can successfully categorize such a population. However, we need to consider the effect of the constant features on the category recovery procedure. In particular, how does the presence of the constant features affect the components of the categorization evaluation function U?

It can be shown that constant features have *no* effect on either U_P or U_C.[9] Therefore, if we treat the genus members as separate population, we

[9]First, consider U_P. Constant features have no uncertainty: $1 \cdot \log 1 = 0$. Thus, U_P is unaffected by the addition of constant features. Next consider U_C. At first one might expect the addition of constant features to add to category uncertainty: constant features are structure shared by all objects, leading to category confusion. We can show that this is *not* the case.

Let us assume we have a world c classes, and that objects are described by m modal

Start:	GS12-27	GS11-2	GS12-17	GS13-38	GS12-20	GS12-32	GS12-21
GS13-46:	GS12-17	GS11-2	GS12-27	GS12-21	GS12-20	GS13-38	GS13-46 GS12-32
GS11-11:	GS11-11	GS12-17	GS11-2	GS12-27	GS12-21	GS12-20	GS13-38 GS13-46 GS12-32
GS13-33:	GS11-11	GS12-17	GS11-2	GS12-27	GS12-21	GS13-33	GS12-20 GS13-38 GS13-46 GS12-32
GS11-13:	GS11-13 GS12-21	GS11-11	GS12-17	GS11-2	GS12-27	GS13-33	GS12-20 GS13-38 GS13-46 GS12-32
GS13-48:	GS11-13 GS12-21	GS11-11	GS12-17	GS11-2	GS12-27	GS13-48 GS12-32	GS13-33 GS12-20 GS13-38 GS13-46
GS12-23:	GS11-13 GS12-21	GS11-11	GS12-17	GS11-2	GS12-27	GS12-23 GS13-46	GS13-48 GS12-32 GS13-33 GS12-20 GS13-38
GS13-37:	GS13-37 GS12-27	GS11-13 GS12-21	GS11-11	GS12-17	GS11-2	GS12-23 GS13-46	GS13-48 GS12-32 GS13-33 GS12-20 GS13-38
GS11-16:	GS12-23 GS12-20 GS12-32	GS13-48 GS13-38	GS13-33 GS13-46	GS12-27 GS12-17	GS11-11	GS12-21	GS11-13 GS13-37 GS11-16 GS11-2
GS12-29:	GS12-23 GS12-20 GS12-32	GS13-48 GS13-38	GS13-33 GS13-46	GS12-29 GS12-21	GS12-27 GS12-17	GS11-11	GS11-13 GS13-37 GS11-16 GS11-2
GS12-26:	GS12-26 GS13-33 GS13-46	GS12-23 GS12-20 GS12-32	GS13-48 GS13-38	GS12-29 GS12-21	GS12-27 GS12-17	GS11-11	GS11-13 GS13-37 GS11-16 GS11-2
GS11-7:	GS12-26 GS13-33 GS13-46	GS12-23 GS12-20 GS12-32	GS13-48 GS13-38	GS12-29 GS12-21	GS12-27 GS12-17	GS11-11	GS11-7 GS11-13 GS13-37 GS11-16 GS11-2
GS12-28:	GS12-28 GS13-48 GS13-38	GS12-26 GS13-33 GS13-46	GS12-23 GS12-20 GS12-32	GS12-29 GS12-21	GS12-27 GS12-17	GS11-11	GS11-7 GS11-13 GS13-37 GS11-16 GS11-2
GS11-10:	GS12-28 GS13-48 GS13-38	GS12-26 GS13-33 GS13-46	GS12-23 GS12-20 GS12-32	GS12-29 GS12-21	GS12-27 GS12-17	GS11-11	GS11-10 GS11-7 GS11-13 GS13-37 GS11-16 GS11-2
GS13-43:	GS13-43 GS12-23 GS12-20 GS12-32	GS12-28 GS13-48 GS13-38	GS12-26 GS13-33 GS13-46	GS12-29 GS12-21	GS12-27 GS12-17	GS11-11	GS11-10 GS11-7 GS11-13 GS13-37 GS11-16 GS11-2

* * *

GS12-31:	GS13-35 GS13-36 GS13-44 GS13-47 GS12-28 GS13-48 GS13-38	GS13-42 GS13-40 GS13-39 GS13-45 GS12-26 GS13-33 GS13-46	GS12-30 GS13-41 GS13-34 GS13-43 GS12-23 GS12-20 GS12-32	GS12-31 GS12-22 GS12-27 GS12-17	GS12-18 GS12-25 GS11-11	GS12-19 GS12-29 GS12-21	GS11-8 GS11-1 GS11-3 GS11-15 GS11-4 GS11-6 GS11-14 GS11-10 GS11-7 GS11-13 GS13-37 GS11-16 GS11-2

Figure 6.9: Recursively categorizing the first genus of the simulated population described in Table 6.3 with λ equal to .65.The species will be correctly recovered.

expect that the categorization algorithm to be able to recover the species. We test this conclusion by executing the categorization algorithm on the members of the first genus of the simulation example of the previous section; Figure 6.9 displays the results of two categorization attempts, with λ set to .65 in both cases. Notice that the correct species categories are recovered; the isolated errors would be corrected in subsequent viewing of the incorrectly categorized objects. Repeated application of the categorization algorithm shows this categorization to be stable. Thus, the observer can reliably recover the species categories once the genera have been separated.

We can test this procedure in the domain of the bacteria as well. Figures 6.10 and 6.11 display the results of re-categorizing the genera *bacteroides* and *fusobacterium*, respectively. For the *bacteroides* the value of λ is .6; for *fusobacterium*, .5. (In a moment, we will discuss the effect on λ caused by recursively categorizing the genera.) In both of these cases the categorization procedure reliably recovers the species. In the previous section, we demonstrated the ability of the categorization procedure to recover the genera. Thus, using a recursive categorization strategy, the observer could recover both the genera *and* the species.

Conceptually, there is a problem with performing recursive categoriza-

---

features (which take a different value for each of the c classes) and by x constant features. Now, let us evaluate the category uncertainty U_C for the categorization corresponding to the modal classes, the "correct" categorization. To do so requires computing the category uncertainty of each object for each possible feature subset description. We know that there are 2^{m+x} possible feature subsets (for this analysis we must include the null set). Because the m features are modal, if any of those features are included in the description of an object, then there is no category uncertainty. If however, there are no modal features in the description, then the object may match any category and the uncertainty is $\log c$. The number of feature subsets containing no modal features is 2^x. Thus the average category uncertainty of each object (and thus for complete average) is:

$$U_C = \frac{2^x}{2^{x+m}} \cdot \log c = \frac{1}{2^m} \cdot \log c$$

That is, U_C is independent of x. The intuition behind the result is that the same *proportion* of feature subsets contain no modal information, regardless of the number of constant features. When there are no constant features, then only the null subset produces category uncertainty. As constant features are added, the number of possible subsets increases by the same ratio as the number of non-modal subsets (namely 2^x). This is true for any categorization we evaluate; we used the correct modal categorization only to make the an analytic computation of the category uncertainty possible.

Start:	BV-32 BF-1 BV-35 BF-6 BV-43 BT-21 BF-13		
BT-24:	BT-24 BV-32 BV-43 BV-35		BF-1 BT-21 BF-13 BF-6
BT-17:	BV-32 BV-35 BF-6 BV-43 BF-13		BT-24 BT-21 BT-17 BF-1
BT-27:	BV-32 BV-35 BF-6 BV-43 BF-13		BT-27 BT-24 BT-21 BT-17 BF-1
BV-40:	BV-40 BV-32 BV-35 BF-6 BV-43 BF-13		BT-27 BT-24 BT-21 BT-17 BF-1
BF-14:	BF-14 BV-40 BV-32 BV-35 BF-6 BV-43 BF-13		BT-27 BT-24 BT-21 BT-17 BF-1
BV-33:	BV-33 BF-14 BV-40 BV-32 BV-35 BF-6 BV-43 BF-13		BT-27 BT-24 BT-21 BT-17 BF-1
BV-31:	BV-31 BV-33 BF-14 BV-40 BV-32 BV-35 BF-6 BV-43 BF-13		BT-27 BT-24 BT-21 BT-17 BF-1
BV-41:	BT-27 BT-24 BT-21 BT-17 BF-1	BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BV-40 BF-6 BF-13 BF-14
BV-39:	BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BV-40 BF-6 BF-13 BF-14
BT-25:	BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BV-40 BF-6 BF-13 BF-14
BF-3:	BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-3 BV-40 BF-6 BF-13 BF-14
BT-23:	BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-3 BV-40 BF-6 BF-13 BF-14
BT-16:	BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-3 BV-40 BF-6 BF-13 BF-14
BT-19:	BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-3 BV-40 BF-6 BF-13 BF-14
BF-7:	BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-7 BF-3 BV-40 BF-6 BF-13 BF-14
BT-22:	BT-22 BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-7 BF-3 BV-40 BF-6 BF-13 BF-14
BV-44:	BT-22 BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-44 BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-7 BF-3 BV-40 BF-6 BF-13 BF-14
BF-9:	BT-22 BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BV-44 BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35	BF-9 BF-7 BF-3 BV-40 BF-6 BF-13 BF-14

* * *

BF-4:	BF-11 BF-5 BT-29 BT-30 BT-20 BT-22 BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BF-4 BF-12 BF-10 BF-2 BF-15 BF-8 BT-18 BF-3 BF-7 BF-13	BV-42 BV-34 BV-37 BV-45 BV-40 BF-6 BV-44 BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35
BT-28:	BT-28 BF-11 BF-5 BT-29 BT-30 BT-20 BT-22 BT-19 BT-16 BT-23 BT-25 BT-27 BT-24 BT-21 BT-17 BF-1	BF-4 BF-12 BF-10 BF-2 BF-15 BF-8 BF-3 BF-7 BF-13	BV-42 BV-34 BV-37 BV-45 BV-40 BF-6 BV-44 BV-39 BV-31 BV-43 BV-32 BV-41 BV-33 BV-35

Figure 6.10: Re-categorizing a population of the *bacteroides* bacteria. For these executions, the value of λ was 0.6. When the population is restricted to only the genus members, the categorization algorithm reliably recovers the species.

Start:	FN-24 FM-12 FN-27 FN-17 FV-31 FM-15 FN-22			
FM-1:	FM-1 FM-12 FM-15 FN-24	FN-27 FV-31 FN-17 FN-22		
FN-21:	FM-1 FM-12 FM-15 FN-24	FN-21 FN-27 FV-31 FN-17 FN-22		
FV-41:	FM-1 FM-12 FM-15 FN-24	FV-41 FN-21 FN-27 FV-31 FN-17 FN-22		
FM-13:	FM-13 FM-1 FM-12 FM-15 FN-24	FV-41 FN-21 FN-27 FV-31 FN-17 FN-22		
FM-8:	FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FV-41 FN-21 FN-27 FV-31 FN-17 FN-22		
FN-30:	FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-30 FV-41 FN-21 FN-27 FV-31 FN-17 FN-22		
FV-39:	FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-21 FN-27 FV-39 FN-22	FV-41 FV-31 FN-30 FN-17	
FN-25:	FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-25 FN-21 FN-27 FV-39 FN-22	FV-41 FV-31 FN-30 FN-17	
FN-28:	FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-41 FV-31 FN-30 FN-17	
FM-5:	FM-5 FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-41 FV-31 FN-30 FN-17	
FV-37:	FM-5 FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-37 FV-41 FV-31 FN-30 FN-17	
FV-43:	FM-5 FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-43 FV-37 FV-41 FV-31 FN-30 FN-17	
FV-44:	FM-5 FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-44 FV-43 FV-37 FV-41 FV-31 FN-30 FN-17	
FV-35:	FM-5 FM-8 FM-13 FM-1 FM-12 FM-15 FN-24	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-43 FV-35 FV-44 FV-37	FN-17 FN-30 FV-31 FV-41
FM-10:	FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-43 FV-35 FV-44 FV-37	FN-17 FN-30 FV-31 FV-41	FM-5 FM-10 FM-13 FM-12	FM-8 FM-1 FN-24 FM-15
FN-16:	FN-16 FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-43 FV-35 FV-44 FV-37	FN-17 FN-30 FV-31 FV-41	FM-5 FM-10 FM-13 FM-12	FM-8 FM-1 FN-24 FM-15
FM-11:	FN-16 FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-43 FV-35 FV-44 FV-37	FN-17 FN-30 FV-31 FV-41	FM-11 FM-5 FM-10 FM-13 FM-12	FM-8 FM-1 FN-24 FM-15
FN-20:	FN-16 FN-28 FN-25 FN-21 FN-27 FV-39 FN-22	FV-43 FV-35 FV-44 FV-37	FN-20 FN-17 FN-30 FV-31 FV-41	FM-11 FM-5 FM-10 FM-13 FM-12	FM-8 FM-1 FN-24 FM-15

* * *

FM-12:	FN-23 FN-26 FN-19 FN-18 FV-38 FN-29 FN-16 FN-28 FN-25 FN-21 FN-27 FV-39 FN-22 FN-17 FN-20 FN-30	FV-42 FV-40 FV-36 FV-45 FV-41 FV-34 FV-31 FV-33 FV-32 FV-43 FV-35 FV-44 FV-37	FM-12 FM-1 FM-4 FM-9 FM-3 FM-11 FM-5 FM-10 FM-13 FM-12 FM-15 FM-6 FM-14 FM-2	
FM-3:	FN-23 FN-26 FN-19 FN-18 FV-38 FN-29 FN-16 FN-28 FN-25 FN-21 FN-27 FV-39 FN-22 FN-17 FN-20 FN-30	FV-42 FV-40 FV-36 FV-45 FV-41 FV-34 FV-31 FV-33 FV-32 FV-43 FV-35 FV-44 FV-37	FM-3 FM-12 FM-1 FM-4 FM-9 FM-3 FM-11 FM-5 FM-10 FM-13 FM-12 FM-15 FM-6 FM-14 FM-2	

Figure 6.11: Re-categorizing a population of the *fusobacterium* bacteria. For these executions, the value of λ was 0.5. When the population is restricted to only the genus members, the categorization algorithm reliably recovers the species.

tion, and that difficulty is hidden within the computation of the normalization factor η. By restricting the population to only one genus, the property uncertainty of the coarsest categorization — the categorization consisting of only one category — is greatly reduced, thereby reducing the value of η. As shown in the equation for U:

$$U(\mathcal{Z}) = (1 - \lambda)\, U_P(\mathcal{Z}) + \lambda\, \eta(\mathcal{Z})\, U_C(\mathcal{Z}) \tag{6.3}$$

η directly multiplies the U_C term, and thus controls (along with λ) the relative contribution of the two uncertainties. [10] Thus the effect of λ on the relative contribution of U_P and U_C to U is modified when η is recomputed. This change in the effect of λ complicates the interpretation of λ as being a trade-off between uncertainties established according to the goals of the observer.

Perhaps, then, within the present context of categorizing objects we should view λ as a parameter controlled by the observer and used by him to probe the category structure of the population. If he can discover category structure that is stable over a range of λ, then he can assert that these categories are more likely to correspond to natural processes. As yet, the question of directly relating the goals of the observer to the categorization process has not been satisfactorily resolved. The intuition that the goals of the observer specify the trade-off bewteen property uncertainty and category uncertainty is strong; more work is needed to re-examine the form of equation 6.3 to resolve this issue.

6.3.2 Primary process requirement

As a final comment about process separation we note that we were able to recover the species structure only after recovering the genera. For recursive categorization to be effective, there must be a *primary process* that can be recovered at each step. In the case of the bacteria, the genera represented

[10] Unlike decreasing λ, lowering the value of η reduces the weight accorded U_C, *without increasing the weight of* U_P. Normally, reducing λ increases the weight of U_P making the categorization evaluation function more sensitive to property variation. This difference, along with some implementation details about local category formation, is the reason that reason that simply lowering λ will not accomplish the separation of the species, but that reducing η will.

a strong modal structure that could be recovered immediately. In the simulation, we provided more modal features for the genus categories than for the species. This imbalance provided a primary process that could initialize the recursive categorization procedure. Therefore, if the observer categorizes objects in a multiple modal world by applying the categorization algorithm recursively, then he must be provided a with a representation that permits him to discover a primary process at each level.

6.4 Evaluating and Assigning Features

The emphasis of this chapter, indeed this thesis, has been on the recovery of natural object categories. We have demonstrated that the observer must be provided with a representation that reflects the modal structure present the world. However, it would be desirable for the observer to be able to *improve* his representation as he categorizes the objects in the world. By "improve," we mean make the representation more sensitive to the natural categories present. In the context of property vectors this process would include "growing" new features that are constrained by the natural modes, as well as assigning computed features to the correct modal level. In this section we provide a mechanism by which the observer can evaluate the effectiveness of features in terms how well they support the recovery of natural mode categories.

6.4.1 Leaves example

Let us return to the leaves example introduced at the start of the chapter. We assume that the categorization algorithm has been executed and the correct natural mode categories — the species — have been discovered. Furthermore, we assume that the observer has determined that no other modal levels exist. Now, we wish to develop a method by which the observer can determine which features are most sensitive to the species categories.

We proceed by creating a short taxonomy of the leaves shown in Figure 6.12. The only levels present are those which correspond to solutions to the categorization algorithm. The middle level corresponds to the species, and is the "natural" level of the taxonomy. This level has as its superordinate the single-category, which is the recovered solution for λ near 1.0.

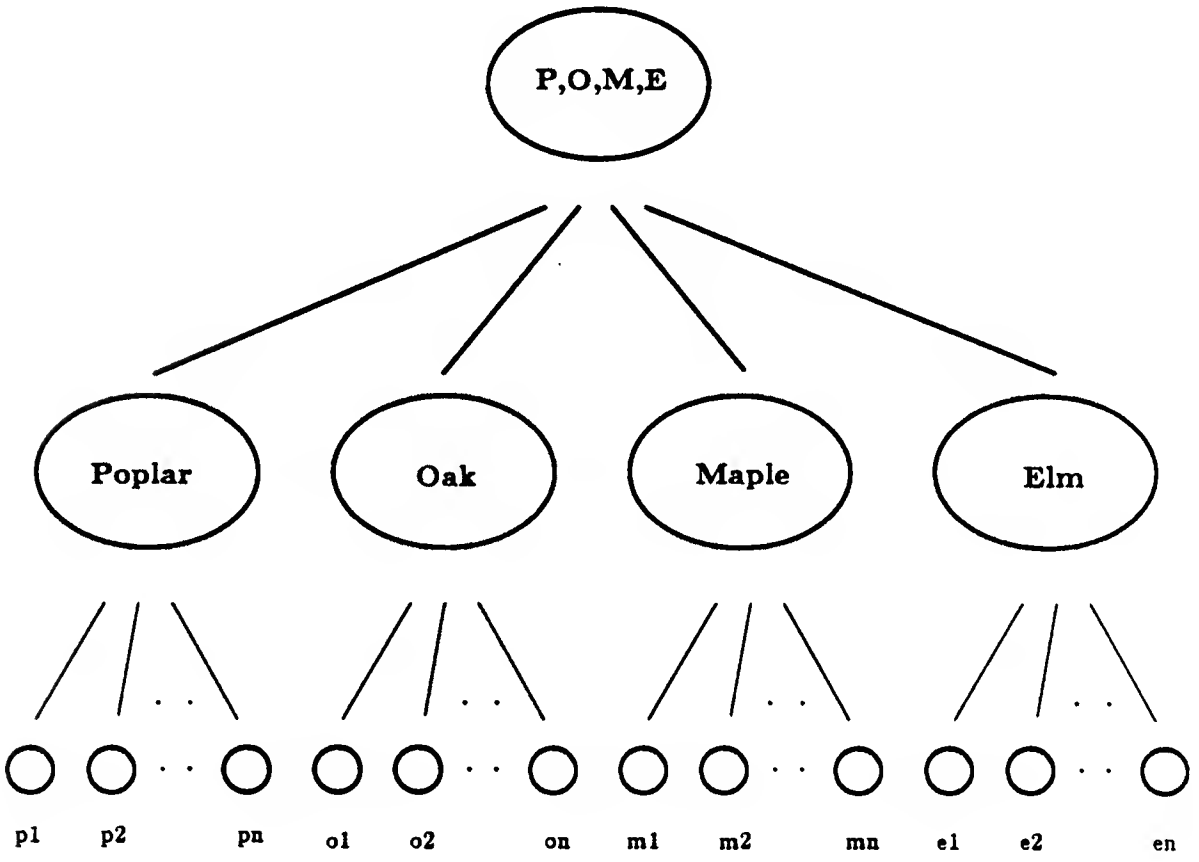


Figure 6.12: A short taxonomy of leaves consisting only of the natural (species) level and the two extreme (or noise) levels above and below.

The level below the species consists of the finest possible categorization, the solution for λ near zero. We construct this taxonomy for the purpose of evaluating the categorization uncertainty measure U for a range of λ and for different subsets of the features. That is, for a given subset of features, we will measure the range of λ for which the species categorization is the preferred level of the taxonomy. We refer to the extent of this range as the λ-*stability* of the features for these natural categories. We will use the λ-stability of the features to evaluate their utility in recovering the category structure.

Tables 6.4 and 6.5 display the λ-stability values for different subsets of the features. (Not all subsets are displayed.) For example, panel (a) of the first table reveals that if only the feature "apex" is used to describe the leaves, then for a range of λ of .87, the species level of the taxonomy is preferred over the other two levels. Notice the inclusion of the feature **nf1**, a noise feature. This feature, whose value is assigned randomly for each object, provides a baseline against which to compare other features. Panel (a) shows all 1-feature subsets, and the λ-stability value associated with each subset. Notice that "apex," "base," and "color" have a relatively high stability, indicating they are the best individual features. This does *not* mean that each of these features is sufficient for the recovery of the species categories; for example, poplar and elm both have a rounded base. Rather, given the particular taxonomy of Figure 6.12 these features are highly selective of the species level. Notice that "width" and "length" have low λ-stability values, indicating little diagnosticity for the species.. Finally, the noise feature has no (significant) λ-stability.

Panel (b) displays some of the 2-feature subsets, including some pairs formed by combining a good single feature with noise. First, notice that the best pair is in fact the combination of the two best single features. This does not have to be the case. For example, suppose the two best single features provide redundant information, and that two other feature are orthogonal in their separation of the population. In that case, the combination of the two orthogonal features would provided a greater separation of the classes and thus a larger λ-stability value. The less the features in a domain dsiaply this form of interaction, the easier it is to evaluate addtional features. The reason for this is that the combinatorics of $\binom{n}{k}$ normally preclude evaluating all possible subsets of features. If the features combine independently, small subsets of features can be tested and then combined.

(APEX)	0.87
(BASE)	0.87
(COLOR)	0.87
(MARGIN)	0.72
(LOBES)	0.68
(FLARE)	0.30
(LENGTH)	0.21
(WIDTH)	0.00
(NF1)	0.00

(a) 1 feature subsets

(BASE COLOR)	0.91
(APEX BASE)	0.91
(APEX COLOR)	0.90
(MARGIN APEX)	0.81
(LOBES APEX)	0.81
(LOBES COLOR)	0.81
(MARGIN BASE)	0.80
(LOBES BASE)	0.78
(MARGIN COLOR)	0.77
(LOBES MARGIN)	0.75
(FLARE COLOR)	0.62
(FLARE BASE)	0.61
(FLARE APEX)	0.58
(FLARE MARGIN)	0.57
(LENGTH BASE)	0.53
(FLARE LOBES)	0.52
(WIDTH APEX)	0.51
(LENGTH LOBES)	0.51

$\sim$

(WIDTH MARGIN)	0.40
(LENGTH FLARE)	0.25
(WIDTH FLARE)	0.24

$\sim$

(COLOR NF1)	0.22
(BASE NF1)	0.22
(LENGTH WIDTH)	0.22
(APEX NF1)	0.20
(LOBES NF1)	0.19
(MARGIN NF1)	0.18
(FLARE NF1)	0.00
(WIDTH NF1)	0.00
(LENGTH NF1)	0.00

(b) 2 feature subsets

(APEX BASE COLOR)	0.93
(MARGIN APEX BASE)	0.86
(MARGIN APEX COLOR)	0.86
(MARGIN BASE COLOR)	0.85
(LOBES APEX COLOR)	0.84
(LOBES BASE COLOR)	0.82
(LOBES APEX BASE)	0.81
(LOBES MARGIN APEX)	0.81

$\sim$

(FLARE APEX COLOR)	0.73
(FLARE APEX BASE)	0.71
(WIDTH APEX BASE)	0.71
(WIDTH APEX COLOR)	0.70
(FLARE BASE COLOR)	0.70
(FLARE LOBES COLOR)	0.69
(FLARE MARGIN BASE)	0.68
(FLARE MARGIN APEX)	0.67
(FLARE MARGIN COLOR)	0.67
(WIDTH BASE COLOR)	0.67

$\sim$

(LENGTH LOBES APEX)	0.56
(WIDTH MARGIN COLOR)	0.56
(LENGTH LOBES MARGIN)	0.56
(LENGTH LOBES BASE)	0.55
(LENGTH MARGIN COLOR)	0.52
(WIDTH FLARE APEX)	0.50
(WIDTH FLARE BASE)	0.50
(WIDTH FLARE COLOR)	0.49
(APEX BASE NF1)	0.45
(WIDTH FLARE LOBES)	0.45
(BASE COLOR NF1)	0.44
(APEX COLOR NF1)	0.44
(LOBES COLOR NF1)	0.44
(LOBES APEX NF1)	0.43
(WIDTH FLARE MARGIN)	0.43
(LENGTH FLARE BASE)	0.43
(LENGTH FLARE COLOR)	0.43
(MARGIN BASE NF1)	0.42
(LENGTH WIDTH BASE)	0.42

$\sim$

(LENGTH APEX NF1)	0.18
(LENGTH WIDTH FLARE)	0.18
(WIDTH MARGIN NF1)	0.16
(LENGTH COLOR NF1)	0.16
(FLARE LOBES NF1)	0.15
(LENGTH MARGIN NF1)	0.14
(LENGTH LOBES NF1)	0.14
(WIDTH FLARE NF1)	0.00
(LENGTH FLARE NF1)	0.00
(LENGTH WIDTH NF1)	0.00

(c) 3 feature subsets

Table 6.4: Selected λ-stability measurements for different subsets of features in the leaves example; subsets of length 1, 2, and 3 are shown. By comparing the addition of a new feature with the addition of a noise feature (**nf1**) one can judge the utility of the new feature.

(MARGIN APEX BASE COLOR)	0.90
(LOBES APEX BASE COLOR)	0.90
(LOBES MARGIN APEX BASE)	0.87
(LOBES MARGIN APEX COLOR)	0.86
(LOBES MARGIN BASE COLOR)	0.84
(FLARE APEX BASE COLOR)	0.82
$\sim$	
(APEX BASE COLOR NF1)	0.64
*	
*	
*	

(a) 4 feature subsets

(LOBES MARGIN APEX BASE COLOR)	0.89
(FLARE MARGIN APEX BASE COLOR)	0.82
(FLARE LOBES APEX BASE COLOR)	0.82
(WIDTH MARGIN APEX BASE COLOR)	0.79
(FLARE LOBES MARGIN APEX BASE)	0.78
(FLARE LOBES MARGIN APEX BASE)	0.78
$\sim$	
(NF1 MARGIN APEX BASE COLOR)	0.69
*	
*	
*	

(b) 5 feature subsets

(FLARE LOBES MARGIN APEX BASE COLOR)	0.87
(WIDTH LOBES MARGIN APEX BASE COLOR)	0.80
(LENGTH LOBES MARGIN APEX BASE COLOR)	0.80
$\sim$	
(NF1 LOBES MARGIN APEX BASE COLOR)	0.71
*	
*	
*	

(c) 6 feature subsets

(WIDTH FLARE LOBES MARGIN APEX BASE COLOR)	0.78
(LENGTH FLARE LOBES MARGIN APEX BASE COLOR)	0.77
$\sim$	
(NF1 FLARE LOBES MARGIN APEX BASE COLOR)	0.64
*	
*	
*	

(d) 7 feature subsets

Table 6.5: Selected λ-stability measurements for different subsets of features in the leaves example; subsets are of length 4, 5, 6, and 7.

Second, note that the subset (width, apex) has a much greater stability value than (apex, nf1). This result demonstrates that although the "width" feature by itself provides little support for the recovery of the natural categories, it does not act as destructively as pure noise. The highly destructive action of the noise feature can be further demonstrated in panel (c) of Table 6.4. Compare the triplet (apex, base, nf1) with both the first triplet of (apex, base, color) and with the top pair in panel (b) of (apex, base). The extreme reduction in the stability caused by the noise is an indication to the observer that the feature "nf1" has little use and should be removed from his representation. By using this comparison to noise strategy, the observer can also evaluate the addition of new features. Given a proposed new feature,[11] this mechanism provides a means to evaluate its utility.

The panels of Table 6.5 show that the maximum value of λ-stability remains quite high (about .9) until the inclusion of the last 3 features "width," "length," and "nf1." Including either "width" or "length" reduces the λ-stability to .78, and the inclusion of the noise feature reduces the stability to a value of .71. Thus, most of the leaf features are relatively uniform in their sensitivity to the species structure of the leaves.

6.4.2 Bacteria example

The fact that the different features of the leaves do not exhibit large differences in their diagnosticity for the species is not surprising; these features were chosen from a leaf identification reference [Preston, 1976]. There is only one modal level and each of the features was chosen to be useful in identifying that level. For the bacteria example, however, there are two modal levels. Some features are sensitive to the genera, others to the species. Therefore let us consider evaluating the features of the bacteria domain.

The taxonomy we construct resembles that for the leaves example, but it includes both natural levels (Figure 6.13). In this case we will evaluate the λ-stability for each natural clustering. Panel (a) of Table 6.6 displays the λ-stability values for several feature subsets for the genus level of the

[11] An important, and open, question is how does the observer propose new features? The literature is quite sparse in addressing this question, with most attempts being confined to arithmetic combinations of previous features (for example see Boyle [19??]). Recently, Michalski and his colleagues have explored the issue of the logical manipulation of features [Michalski, 19??].

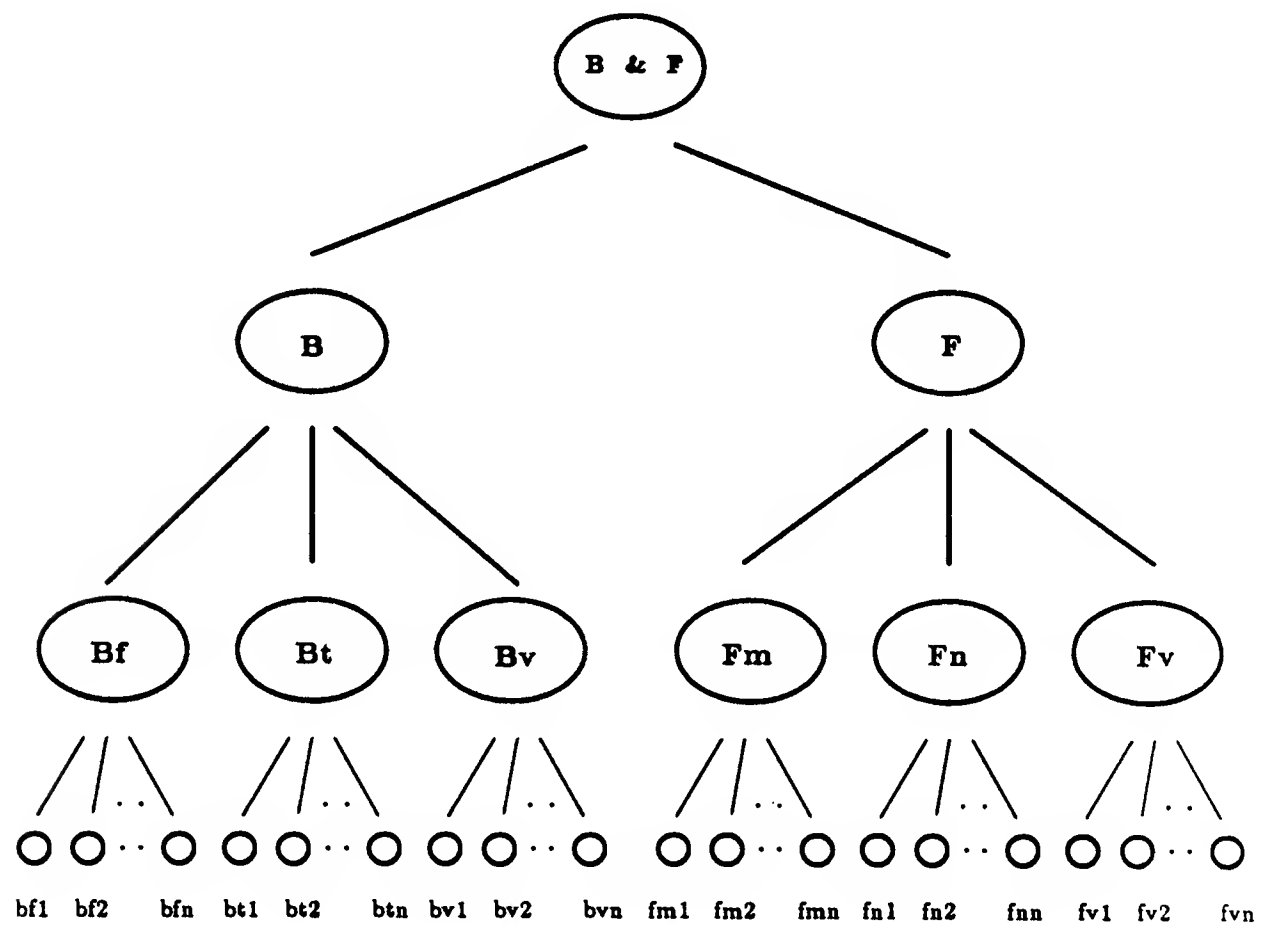


Figure 6.13: A taxonomy of the bacteria of Table 6.2 separated according to genus and species.

(loc gr-kan glc)	1.00
(loc gr-pen glc)	0.82
(gr-pen gr-kan glc)	0.81
(loc gr-pen gr-kan)	0.81
(gr-rif gr-kan glc)	0.63
~	
(loc gr-kan nfl)	0.60
*	
*	
*	

3 feature subsets

(loc gr-kan glc gr-pen)	0.85
(loc gr-kan glc gr-rif)	0.71
~	
(loc gr-kan glc nfl)	0.67
*	
*	
*	

4 feature subsets

(loc gr-kan gr-pen glc gr-rif)	0.72
~	
(loc gr-kan gr-pen glc nfl)	0.66
*	
*	
*	

5 feature subsets

(a)

(rham)	0.87
(esculin)	0.87
(dole)	0.87
(nfl)	0.00

1 feature subset

(rham esculin)	0.90
(esculin dole)	0.90
(rham dole)	0.90
(rham nfl)	0.12

2 feature subsets

(rham escuin dole)	0.93

3 feature subsets

(b)

(bile esculin gr-rif)	0.90
(esculin dole gr-rif)	0.89
(bile dole gr-rif)	0.89
(bile esculin dole)	0.89
(dole gr-rif gr-pen)	0.59
(bile dole gr-pen)	0.58
(bile esculin gr-pen)	0.58
*	
*	
*	

3 feature subsets

(bile esculin dole gr-rif)	0.93
(esculin dole gr-rif gr-pen)	0.70
*	
*	
*	

4 feature subsets

(bile esculin dole gr-rif gr-pen)	0.76
*	
*	
*	

5 feature subsets

(c)

Table 6.5: Evaluation of features for the separation of the (a) bacteria genera and of the species (b) *bacteroides* and (c) *fusobacterium*.

bacteria taxonomy; included are the best feature subsets of length 3,4, and 5. Notice that the best subset of length 3 (location, gr-kan, glc) has a λ-stability value of 1.0; this maximum value occurs because these features are modal for the different genera (see Table 6.2). The best subset of length 4 adds the feature "gr-pen," but this lowers the stability value to .86. Notice that second best subset of length 4 reduces that value to .71. Finally, the best subset of length 5, generated by including "gr-rif," is only marginally better than a subset generated by adding a noise feature to the best length 4 subset (.72 as opposed to .66). These results demonstrate that the features (location, gr-kan, glc, gr-pen) are the features most constrained by the processes associated with the genus of the bacteria, and that the other features provide little useful genus information.

Next, we evaluate the separation of species within the genus. For the genus *bacteroides*, the features "gr-rif" and "bile" are constant, providing no information about the species. Thus the only remaining features are "rham," "esculin," and "dole." Since each of these take on one value for one of the species and another value for the other two, and because they each single out a different one of the three species, these three features behave identically with respect to λ-stability. This behavior is indicated in panel (b) of Table 6.6. For the *fusobacterium* genus, however, the features do have a differential effect. As shown in panel (c), the λ-stability remains quite high (about .9) for the best feature subsets of length 4 or less. However, the best subset of length 5 requires the addition of feature "gr-pen" and the λ-stability is greatly reduced (.76). As "gr-pen" was one of the features discovered to be important for the separation of the genera, we know that this feature crosses the modal levels, and thus is a weak feature for the species clustering.

We can summarize the results of the bacteria example by displaying an annotated taxonomy of the domain (Figure 6.14). The features tied to the branches represent the properties of the objects constrained by the natural processes responsible for that particular natural division. In essence, the observer has learned not only to identify the natural categories, but also to relate the properties of objects to to the natural processes in the world.

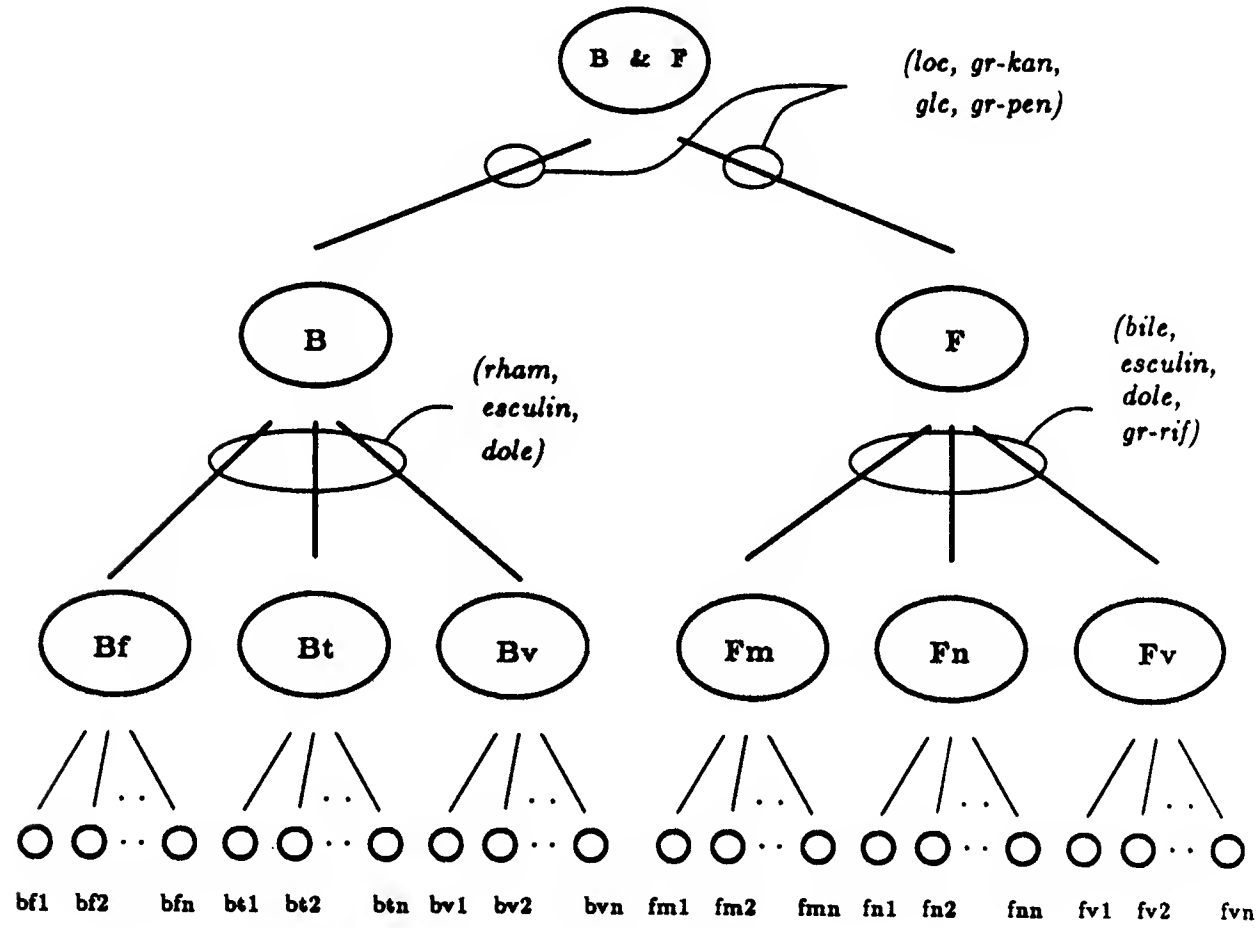


Figure 6.14: The same taxonomy of the bacteria as in Figure 6.13, but annotated with the features that are important in performing the different levels of categorization. These features are constrained by the processes responsible for the different modal levels.

Chapter 7

Conclusion

7.1 Summary

We began this thesis with the following three questions:

- What are the necessary conditions that must be true of the world if a set of categories is to be useful to the observer in predicting the important properties of objects?
- What are the characteristics of such a set of categories?
- How does the observer acquire the categories that support the inferences required?

Let us consider each in turn.

The first question is about the *world.* What needs to be special about the world if the observer is to be able to make inferences about the important properties of objects? As an answer, we proposed the Principle of Natural Modes: the interaction between the processes that create objects and the environment that acts upon them causes objects to cluster in the space of properties important to their interaction with the environment. The importance of this claim is that without such a constraint, many of the perceptual inferences that are necessary to the survival of the observer cannot be made. This statement is true even at the lowest levels of perception. For example, consider the method by which a tick finds a host. The tick climbs onto a branch or blade of tall grass, waits until it detects the presence of buteric acid

(a chemical produced by warm blooded animals), then releases the branch (or jumps) and falls towards the ground. If no host is underneath, the tick starts again. Now let us consider the tick's strategy in terms of natural modes. All mammals have many biological processes in common that are unique to mammals; as such, mammals form a natural mode. The tick's strategy is an effective one because the presence of buteric acid is strong indicator of the proximity of a mammal. Although one can view this inference simply as a high correlation statistic, the underlying reason why the strategy of the tick is successful is because buteric acid is a good predictor of an object belonging to the natural mode of mammal.

Although the inferences that must be made by a human observer may be more varied and more complex than those of a tick, the principles underlying the predictions of unobserved properties are no different. Given an apple, we know we can eat it. Given a tiger, we know to run. The necessary requirement for being able to make these inferences is that we must be able to determine the natural mode to which an object belongs. The categories we use to describe these objects must be consistent with the natural mode structure of the world.

The existence of natural modes allows us to define the problem of categorization, namely the recovery of object categories corresponding to the natural modes important to observer. Our solution to this problem required decomposing the task into two components. First, the observer must be able to identify when a set of categories corresponds to natural classes, and second, he must be able to recover such a set of categories from the available data. These two components provide the answers to the second and third questions this thesis sought to address.

We constructed a measure of how well a set of categories reflected the natural modes by measuring how well the categories supported the inference requirements of the observer. We argued that if a set of categories satisfied the goals of the observer and permitted him to make the necessary inferences about the properties of objects, then that set of categories must capture the structure of the natural modes.

In our analysis of the goals of the observer and of the characteristics of a set of categories that support those goals, we identified two conflicting constraints. First, the observer requires that knowledge of the category of an object be sufficient to make strong inferences about the properties and behavior of that object. This requirement favors the formation of fine, ho-

mogeneous categories. Such categories are highly structured and thus convey much information about the properties of their members. Larger categories have less constrained properties and thus the observer has a greater *property uncertainty* once the category of an object is known. Second, the inferences made by the observer must be reliable; thus, he requires that the assignment of an object to a category be a robust process. Such a constraint favors the formation of coarse categories, where few properties are needed to determine category membership. The coarser a set of categories, the easier it is to determine category membership of an object; there is less *category uncertainty* for given object.

Therefore, the observer is faced with a trade-off between the ease of making an inference about an object and the specificity of the inference. To make this trade-off explicit, we derived a measure for each of the two uncertainties (based on information theory) and combined them using a free parameter λ as a relative weight. This combined measure — referred to as the total uncertainty of a categorization — allowed the observer to explicitly control the balance of uncertainty. If the observer requires precise inferences, a low value of λ selects tightly constrained categories; these categories provide the necessary inferential power, but at the expense of requiring detailed information about an object to determine the category to which it belongs. Likewise, if the observer requires a robust categorization procedure even when little sensory information is provided, a high value of λ causes coarse categories to be preferred; they are easily accessed with little sensory information, but they permit the observer to make only weak inferences about the properties of objects.

The measure we derived is based solely on the goals of the observer; sets of categories which support the goals of inference of the observer yield lower total uncertainty than those that do not. But how do we relate this measure to the natural modes? As argued above, we know that the goals of inference of the observer can only be accomplished if he recovers the natural categories. It is the structure of the modes that permits the inference of unobserved properties from observed properties. Thus, by directly measuring how well the observer accomplishes his goal, we are measuring how successful he has been in recover the natural categories.

Having constructed a measure capable of evaluating the degree to which a set of categories captured the natural mode structure, we next considered the problem of recovering the natural categories from the data provided

by the environment. Based upon the learning paradigm of formal learning theory, we defined a *categorization paradigm* that allowed us to identify the components of the categorization process. This paradigm was developed to be consistent with our intuitions about the categorization procedure; for example, objects are viewed sequentially, with the observer modifying his current categorization of objects as each new data item is viewed.

There are three critical components in the paradigm. First, is the *representation,* the information encoded by the observer upon viewing an object. If the representation does not have the power to distinguish between objects in different natural modes — if the representation is not *class preserving* — then the observer can not hope to recover the natural categories. Second, is the *hypothesis evaluation function,* which provides the criteria by which the observer chooses a particular categorization. The last component of the paradigm consists of of a *hypothesis generation method.* This component, which is responsible for producing categorization hypotheses, is also critical to the success of the categorization procedure. Because of the combinatorics of partitioning problem, one can not attempt all possible categorizations in a world of many objects. Therefore, one needs to develop a procedure that will eventually converge to the correct set of categories.

Using the paradigm as a model we constructed a categorization procedure. This procedure implements the total uncertainty of a categorization as the hypothesis evaluation function. The hypothesis generation method we present is a dynamic, data driven procedure. Upon viewing a new object, the observer produces a new set of categories by modifying the previous hypothesis. Although the algorithm is statistical in nature, and not *guaranteed* to produce the correct categorization, we have demonstrated its effectiveness in several domains. One of these domains consists of the soybean data of Michalski and Stepp [1983], which have been shown to be challenging for standard clustering techniques. The algorithm successfully recovered the four species of diseases present and did not require the a priori knowledge of the number of classes contained in the population.

Finally, we considered the case of a *multiple mode* domain, a domain in which there is more than one level of natural structures. The example we used was that of infectious bacteria, where there is structure at both the genus and species level. We first demonstrated a technique by which the observer could recover the different levels present. This technique relies on the observer being able to detect a primary process level; once the

observer discovers these categories he can then recursively categorize each sub-population in search of additional structure. To support such a procedure we analyzed the case of attempting to categorize a world in which there is no modal structure. By determining the pathological behavior observed in such a situation, we provide the observer with the necessary halting conditions for the recursive strategy.

An important aspect of the multiple mode analysis was the development of a method for evaluating the utility of a feature for recovering the natural categories. In the single mode world, this technique provides the observer with the means for evaluating new features, and thus permits him to learn a better representation. In the multiple modal world this technique also provides a mechanism for assigning the different features encoded by the observer to the different process levels present in the domain. This technique begins to address the fundamental problem of recovering natural *processes* as opposed to recovering only the categories formed by the processes.

7.2 Clustering by Natural Modes

One of the contributions of this work is a new method by which to measure the quality of a set of categories. The measure U — the total uncertainty of a categorization — reflects how well the categories support the goals of making inferences about the properties of objects. How does this method compare to other clustering techniques?

First, we again mention that the categorization procedure based upon the uncertainty measure was capable of successfully categorizing the soybean data of Michalski and Stepp[1983]. In their work, they report experiments in which they attempted to categorize those data using 18 different numerical clustering techniques. Of these, only 4 were successful. Thus, for at least this set of data the performance of the categorization technique is at least comparable to other clustering algorithms. Because the uncertainty measure has the desirable property of being insensitive to unconstrained features, it provides a robust method of recovering categories in a domain in which irrelevant features contaminate the data. Furthermore, we have provided a technique by which the relevance of a feature can be assessed once the correct categories are known.

But more important than the performance of the algorithm is the basic

design of the categorization evaluation function. This function explicitly measures how well a particular set of categories supports the goals of making inferences about the properties of objects. Unlike standard techniques that use a distance metric which is assumed to bear some relation to the desired structure of the categories, the uncertainty measure directly evaluates the utility of the categories in terms of the inferences that can be supported. By directly measuring the degree to which a categorization supports the performance of the task of interest (namely that of making inferences), we are more likely to discover a useful set of categories.

7.3 The Utility of Natural Categories: Perception and Language

Throughout this thesis we have motivated the categorization problem by considering the inference requirements of the observer. However, other problems of cognitive science are made less severe if the cognitive system can recover the structure of the natural world. In particular, let us return to Quine's question of natural kinds ([Quine, 1969] and section 2.3.1). Quine theorized that intelligent communication between individuals is possible only if the individuals share a common description of the world. That is, the similarity space — the *qualia* — of the individuals must be identical, or at least approximate. Without a common descriptive space, the individuals would not be able to resolve the problem of reference: determining the extension in the world of some vocabulary term used or of some gesture made by another individual. In light of this requirement the ability to recover the natural structure in the world provides a basis for communication between individuals. By recovering categories that correspond to natural classes — classes defined by processes in the world — different observers can be assured of convergent world descriptions. If two observers are categorizing a population of objects using identical categorization evaluation functions, and *if the categorization function is appropriate for recovering natural categories* then the two observers are guaranteed to recover similar categories. Therefore, these observers will be able to develop a common description of objects to serve as basis for a mutual vocabulary.

7.4 Recovering Natural Processes: Present and Future Work

The motivation we have presented for this work centers on the task of making inferences about objects. In particular we have argued that the observer must be able to make inferences about unobservable properties of objects given only sensory information. This task led us to the Principle of Natural Modes and to the task of recovering natural object categories.

But making inferences about objects is only a sub-goal of a much more general perceptual goal: understanding the world. The purpose of our perceptual mechanisms is to convey information about the world that is important to our survival. One implication of this goal is that we can improve upon the goal of recovering the natural categories in the world. We know that the natural modes are caused by the interaction of natural processes. Therefore, a more complete understanding of the world is achieved if we recover (discover) the natural processes that are present in the environment.

The last section of chapter 6 — the chapter concerning worlds with multiple natural modes — demonstrated a technique by which the observer could assign the different features to the different process levels present in the domain. In the case of the bacteria, certain features were identified as being constrained by the genus, and others by the species. This capability begins to give the observer an understanding of the natural processes responsible for the natural modes. He does not only acquire the modes themselves, but also gains the knowledge of how the natural processes constrain the properties of objects.

One of the potential extensions to this work is to make explicit the concept of natural processes and attempt to recover the processes directly. We would still assume that classes exist in the world — natural modes. However, we would associate a generating process with each class, responsible for producing all the objects in that class. Now, we change the categorization task by requiring that the observer propose generating processes to explain the observed objects.

An example: Suppose the observer has seen 100 different objects, and his task is to propose generating processes to account for them. As before he could propose a single category, which encompasses all objects. This would correspond to the universal Turing machine process capable of producing

all objects. Alternatively, the observer could propose 100 different generating processes each capable of producing only one object. Just as with the hypothesizing of categories, the observer will want to propose those processes which correspond to the natural modes, which permit him to make inferences about properties of objects. In this case however, the observer has a vocabulary of processes; he must know (or somehow learn) about the types of physical processes that can occur. In other words, he is measuring his uncertainty about properties of the object's generating process as opposed to uncertainty about the properties directly. This approach has much greater power than a simple property vector scheme because the categories are formed by constraint on their physical processes as opposed to constraint on particular properties. And, in the real world, it is the processes that are constrained.

By searching for processes directly, we would reduce the dependence on the property representation. This is a desirable goal given the common belief that no simple set of properties is going to be sufficient for recognition; the general failure of standard pattern recognition techniques supports this opinion. Thus an alternative approach is necessary, and we need to be able to incorporate the ideas and principles developed in this thesis into a more general framework.

References

Abelson, H. and A. diSessa [1981], *Turtle Geometry*, MIT Press, Cambridge, MA.

Anderburg, M. R. [1973], *Cluster Analysis for Applications*, Academic Press, New York.

Ball, G. H. and D. J. Hall [1967], "A clustering technique for summarizing multivariate data," *Behavioral Science,* **12**, 153 – 155.

Beach, L. R. [1964], "Cue probabilism and inference behavior," *Psychological Monographs,* **78**, Whole No. 582.

Bennett, B. M., D. D. Hoffman, and C. Prakash [1987] *Observer Mechanics,* MIT Press, Cambridge, MA.

Bobick, A. and W. A. Richards [1986], "Classifying Objects from Visual Information," MIT Artificial Intelligence Laboratory Memo 879.

Boyle, B. [1974], "Pattern recognition for credit granting," *Proceedings of the National Electronics Conference*, 353–356.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone [1984], *Classification and Regression Trees,* Wadsworth Intl. Group, Belmont, CA.

Carey, S. [1986], "Constraints on Word Meanings – Natural Kinds," *in press.*

Cerella, J. [1979], "Visual classes and natural categories in the pigeon," *J. of Experimental Psychology: Human Perception and Performance,* **5**, 68–77.

Chomsky, N. [1965], *Aspects of the Theory of Syntax*, MIT Press, Cambridge, MA

Dale, M. B. [1985], "On the comparison of conceptual clustering and numerical taxonomy," *IEEE Trans. on Pattern Analysis and Machine Intelligence,* **PAMI-7**, 2, 241–244.

DeJong, G. [1986], "An approach to learning from observation," in Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. [1986], *Machine Learning* —

An Artificial Intelligence Approach, Vloume II Tioga Publishing Co., Palo Alto, CA, 571–590.

Dowell Jr., V. R. and S. D. Allen [1981], "Anaerobic Bacterial Infections," in Balows, A. and Hausler Jr. ,W. J. (eds.) *Diagnostic Procedures for Bacterial, Mycotic and Parasitic Infections,* American Public Health Assoc. Inc., Washington DC, 171–214.

Duda, R. and P. Hart [1973], *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York, NY.

Dumont, J. P. C. and R. M. Robertson, "Neuronal circuits: An evolutionary perspective," *Science,* **233**, 849–853.

Frotier, J. J. and H. Solomon [1966], "Clustering Procedures," in Krishnaiah, P. R. (Ed.), *Multivariate analysis,* Academic Press, New York, 493–506.

Gibson, J. J. [1979], *The Ecological Approach to Visual Perception*, Houghton Mifflin Co., Boston, MA.

Gold, E. M. [1967], "Language identification in the limit," *Information and Control,* **10**, 447–474.

Goodman, N. [1951], *The Structure of Appearance,* Bobs-Merrill Co., New York, NY.

Hand, D. J. [1981], *Discrimination and classification,* John Wiley and Sons, New York, NY.

Hartigan, J. A. [1975], *Clustering Algorithms,* John Wiley & Sons, NY, 14–16.

Hernstein, R. J., D. H. Loveland, and C. Cable [1976], "Natural concepts in pigeons," in *J. of Experimental Psychology: Animal Behavior Processes,* **2**, 285–311.

Hernstein, R. J. [1982], "Objects, categories, and discriminative stimuli," in Roitblat, H. L., Bever, T. G., and Terrace, H. S. (Ed.) *Animal Cognition: Proceedings of the Henry Frank Guggenheim Conf., June, 1982,* Lawrence Erlbaum Assoc., Hillsdale, NJ, 233–261.

Hernstein, R. J. and P. A. de Villiers [1980], "Fish as a natural category for people and pigeons," *The Psych. of Learning and Motivation,* **14**, 59 – 95.

Hoffman, D. and Richards, W. [1986], "Parts of recognition," in A. Pentland (ed.) *From Pixels to Predicates,* Ablex, Norwood, NJ.

Jardine, N. and R. Sibson [1971], *Mathematical Taxonomy*, John Wiley & Sons, London.

Jolicoeur, P., M. A. Gluck, and S .M. Kosslyn,"Pictures and Names: Making

the Connection," *Cognitive Psychology*, **16**, 2, 243–275.
Jones, G. V. [1983], "Identifying basic categories," *Psychological Bulletin,* **94**, 3, 423–428.
Kahneman, D. and Tversky, A [1973], "On the psychology of prediction," *Psychological Review,* **80**, 237–251.
Kass, M. and A. P. Witkin [1985], "Analyzing Oriented Patterns," Schlumberger AI TR–42. Also presented at IJCAI, 1985.
Keil, K. C. [1981], "Constraints on Knowledge and Cognitive Development," *Psychological Review,* **88**, 3, 197–226.
Lance, G. N. and W. T. Williams [1967], "A general theory of classificatory sorting strategies. II. Clustering Systems", *Computer J.*, **10**, 3, 271–277.
Langley, P., G. L. Bradshaw and H. A. Simon [1983], "Rediscovering Chemistry With the BACON System," in *Machine Learning — An Artificial Intelligence Approach,* ed. by R. Michalski, J. Carbonell, and T. Mitchell, Tioga Publishing Co., Palo Alto, CA.
Lebowitz, M. [1986a], "Not the path to perdition: The utility of similarity-based learning," *Proceedings AAAI, 1986,* Philadelphia, PA, 533–537.
Lebowitz, M. [1986b], "Concept learning in a rich input domain: Generalization-based memory," in Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. (eds) *Machine Learning — An Artificial Intelligence Approach, Volume II* Tioga Publishing Co., Palo Alto, CA, 193–214.
Levi, B. [1986], "New Global Formalism Describe Paths to Turbulence," *Physics Today,* **39**, 4, 17–18.
Lozano-Perez, T. [1985], "Shape–from–Function", MIT Artificial Intelligence Lab. Revolving Seminar.
Luenberger, D. G. [1984], *Linear and Non-linear Programming, Second Edition,* Addison-Wesley, Reading, MA.
Marr, D. [1977], "Artificial Intelligence — A Personal View," *Artificial Intelligence,* **9**, 37–48.
Marr, D. [1970], "A theory of cerebral neocortex", *Proc. R. Soc. Lond. B,* **200**, 269–294.
Marr, D. and H. K. Nishihara [1978], "Representation and recognition of the spatial organization of three-dimensional shapes," *Proc. R. Soc. Lond. B,* **200**, 269–294.
Mayr, E. [1984], "Species Concepts and Their Applications" in Sober, E. (ed) *Conceptual Issues in Evolutionary Biology: An Anthology*, MIT Press, Cambridge, 531–541.

McEliece, R. J. [1977], *The Theory of Information and Coding*, Addison-Wesley, Reading, MA.

McMahon, T. A. [1975], "Using body size to understand the structural design of animals: Quadruped Locomotion," *J. of Applied Physiology,* **39**, 619–627.

Michalski, R. S. [1980], "Pattern recognition as rule guided inductive inference," *IEEE Trans on Pattern Analysis,* **2**, 4, 349–361.

Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. [1983], *Machine Learning — An Artificial Intelligence Approach,* Tioga Publishing Co., Palo Alto, CA.

Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. [1986], *Machine Learning — An Artificial Intelligence Approach, Vloume II* Tioga Publishing Co., Palo Alto, CA.

Michalski, R. S. and R. E. Stepp [1983a], "Learning From Observation: Conceptual Clustering," in *Machine Learning — An Artificial Intelligence Approach,* ed. by R. Michalski, J. Carbonell, and T. Mitchell, Tioga Publishing Co., Palo Alto, CA.

Michalski, R. S. and R. E. Stepp [1983b], "Automated cnstruction of classifications: Conceptual clustering versus numerical taxonomy," *IEEE Trans on Pattern Analysis and MAchine Intelligence,* **PAMI-5**, 4, 396–410.

Mitchell, T. M. [1983], "Learning and problem solving," *Proceedings IJCAI-83,* Karlsruhe, West Germany, 1139–1151.

Murphy, L.G. [1982], "Cue validity and levels of categorization," *Psychological Bulletin,* **91**, 1, 174– 177.

Murphy, G. L. and Smith, E. E. [1982], "Basic-level superiority in picture categorization," *J. of Verbal Learning and Verbal Behavior,* **21**, 1–20.

Osherson, D. N. [1978], "Three conditions on conceptual naturalness," *Cognition,* **6**, 263-289.

Osherson, D., M. Stob, and S. Weinstein [1986], *Systems that Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists,* MIT press, Cambridge, MA.

Pentland, A. P. [1984], "Fractal-based description of natural scenes," *IEEE Trans. Pattern Anal. Machine Intellignece,* **6**, 661 – 674.

Pentland, A. P. [1986], "Perceptual Organization and the Representation of Natural Form," Artificial Intelligence, **28**, 293 – 331.

Potter, M. C. and B. A. Faulconer [1975], "Time to understand pictures and words," *Nature (London),* **253**, 437–438.

Preston Jr., R. J. [1976], *North American Trees, Third Edition,* Iowa State Univ. Press, Ames, IA.

Quinlan, J. R. [1986], "Induction of decision trees," *Machine Learning,* **1**, 1.

Reed, S.K. [1972], "Pattern recognition and categorization," *Cognitive Psychology,* **3**, 392–407.

Rosch, E. [1978], "Principle of Categorization," in *Cognition and Categorization,* ed. by E. Rosch and B. Lloyd, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 28–49.

Rosch, E., and Mervis, C. B. [1975], "Family resemblances: Studies in the internal structure of categories," *Cognitive Psychology,* **7**, 573–605.

Rosch, E., Mervis, C. B., Gray, W. D., Johnson, D. M., and Boyes-Braem, P. [1976], "Basic objects in natural categories," *Cognitive Psychology,* **8**, 382–439.

Rota, G. C. [1964], "The number of partitions," *American Math. Mon.,* **71**, 498–504.

Shannon, C. E. and Weaver, W. [1949], *"The Mathematical Theory of Communication,"* Univ. of Illinois Press, Urbana, Illinois.

Smith, E. E. and Medin, D. L. [1981], *Categories and concepts,* Harvard University Press, Cambridge, MA.

Sober, E. (ed) *Conceptual Issues in Evolutionary Biology: An Anthology,* MIT Press, Cambridge, MA.

Stebbins, G. L. and F.J. Ayala [1985], "Evolution of Darwinsim", *Scientific American,* **253**, 1, 74.

Thompson, D. [1961] *On Growth and Form,* Cambridge University Press, Cambridge, Great Britain.

Tversky, A. [1977], "Features of similarity," *Psychological Review,* **84**, 327–352.

Walls, G. L. [1963], *The Vertebrate Eye and Its Adaptive Radiation,* Harper Publishing Co., NY, 519. Originally published in 1942 by Cranbook Institute of Science.

Watanabe, S. [1985], *Pattern recognition: Human and mechanical,* John Wiley and Sons, New York, NY.

Winston, P. H. [1977], *Artificial Intelligence,* Addison-Wesley, Reading, MA.

Winston, P. H. [1979], "Learning and Reasoning by Analogy", MIT Artificial Intelligence Lab. Memo 520. Also appears in abbreviated form in *Comm. ACM*, **23**, 12, 1980.

Winston, P. H., T. O. Binford, B. Katz, and M. Lowry [1983], "Learning

Physical Descriptions from Functional Definitions, Examples, and Precedents," MIT Artificial Intelligence Lab. Memo 679.

Wishart, D. [1969], "Mode analysis: a generalization of nearest neighbor which reduces chaining effect," in Cole, A. J. (ed.) *Numerical Taxonomy,* Academic Press, London, 282–308.

Witkin, A. P. [1983], "Scale-space filtering," *Proceedings IJCAI-83,* Karlsruhe, West Germany, 1019–1022.

Witkin, A. P. and J. M. Tenenbaum [1983], "On the Role of Structure in Vision" in *Human and Machine Vision* ed. J. Beck, B. Hope, and A. Rosenfeld, Academic Press, New York.

Zahn, C. T. [1971], "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Trans. Computers,* **C-20**, 1, 68–87.

Appendix A

Property Specifications

```
;;; These are the specifications for the leaf examples.  Values are generated by random selection from values
;;; list.  Some values are repeated to yield non-uniform distribution

(defvar POPLAR-SPEC (make-instance 'species-specification
                                   :genus 'populus
                                   :species 'tremuloides
                                   :common-name 'poplar
                                   :feature-choice-list
                                   '((length (1.0 2.0 2.0 3.0 ))
                                     (width (1.0 1.0 2.0 ))
                                     (flare (0.0 1.0))
                                     (lobes (1.0 ))
                                     (margin (crenate serrate))
                                     (apex (acute))
                                     (base (rounded))
                                     (color (yellow )))))


(defvar OAK-SPEC (make-instance 'species-specification
                                   :genus 'quercus
                                   :species 'alba
                                   :common-name 'oak
                                   :feature-choice-list
                                   '((length (5.0 6.0 6.0 7.0 7.0 8.0 8.0 9.0))
                                     (width (2.0 3.0 3.0 4.0 4.0 5.0 ))
                                     (flare (-1.0 -2.0))
                                     (lobes (7.0 9.0))
                                     (margin (entire))
                                     (apex (rounded))
                                     (base (cumeate))
                                     (color (light)))))


(defvar COTTON-SPEC (make-instance 'species-specification
                                   :genus 'populus
                                   :species 'deltoides
                                   :common-name 'cotton
                                   :feature-choice-list
                                   '((length (3.0 4.0 4.0 5.0 5.0 6.0 ))
                                     (width (2.0 3.0 3.0 4.0 4.0 5.0 ))
                                     (flare (2.0))
                                     (lobes (1.0))
                                     (margin (crenate))
                                     (apex (acuminate ))
                                     (base (truncate))
                                     (color (yellow light dark )))))
```

```
(defvar MAPLE-SPEC (make-instance 'species-specification   ;;; really sugar maple
                                  :genus 'acer
                                  :species 'saccharum
                                  :common-name 'maple
                                  :feature-choice-list
                                  '((length (3.0 4.0 4.0 6.0))
                                    (width (3.0 4.0 4.0 5.0))
                                    (flare (0.0))
                                    (lobes (5.0))
                                    (margin (entire))
                                    (apex (acute))
                                    (base (truncate))
                                    (color (light)))))


(defvar BIRCH-SPEC (make-instance 'species-specification  ;; really paper birch
                                  :genus 'betula
                                  :species 'papyrifera
                                  :common-name 'birch
                                  :feature-choice-list
                                  '((length (2.0 3.0 3.0 4.0 4.0 5.0))
                                    (width (1.0 2.0 2.0 3.0))
                                    (flare (1.0))
                                    (lobes (1.0))
                                    (margin (doubly-serrate))
                                    (apex (acute))
                                    (base (rounded))
                                    (color (dark)))))

(defvar ELM-SPEC (make-instance 'species-specification
                                  :genus 'ulmus
                                  :species 'americana
                                  :common-name 'elm
                                  :feature-choice-list
                                  '((length (4.0 5.0 5.0 6.0))
                                    (width (2.0 3.0 3.0 ))
                                    (flare (0.0 -1.0))
                                    (lobes (1.0))
                                    (margin (doubly-serrate))
                                    (apex (accuminate))
                                    (base (rounded))
                                    (color (dark)))))
```

```
;;; Soybean specifications.

(defvar SOY-A-SPEC (make-instance 'soy-specification
                                  :common-name 'a*
                                  :feature-choice-list
                                  '((time  (3 4 5 6))
                                    (stand (0))
                                    (precip (2))
                                    (temp (1))
                                    (hail (0))
                                    (years (1 2 3))
                                    (damage (0 1))
                                    (severity (1 2))
                                    (treatment (0 1))
                                    (germ (0 1 2))
                                    (height (1))
                                    (cond (1))
                                    (lodging (0 1))
                                    (cankers (3))
                                    (color (0 1))
                                    (fruit (1))
                                    (decay (1))
                                    (mycelium (0))
                                    (intern (0))
                                    (sclerotia (0))
                                    (pod (0))
                                    (root (0)))))

;;;
(defvar SOY-B-SPEC (make-instance 'soy-specification
                                  :common-name 'b*
                                  :feature-choice-list
                                  '((time  (3 4 5 6))
                                    (stand (0))
                                    (precip (0))
                                    (temp (1 2))
                                    (hail (0 1))
                                    (years (0 1 2 3))
                                    (damage (2 3))
                                    (severity (1))
                                    (treatment (0 1))
                                    (germ (0 1 2))
                                    (height (1))
                                    (cond (1))
                                    (lodging (0 1))
                                    (cankers (0))
                                    (color (3))
                                    (fruit (0))
                                    (decay (0))
                                    (mycelium (0))
                                    (intern (2))
                                    (sclerotia (1))
                                    (pod (0))
                                    (root (0)))))
```

```
;;;
(defvar SOY-C-SPEC (make-instance 'soy-specification
                                  :common-name 'c*
                                  :feature-choice-list
                                  '((time  (0 0 2 2 3 4))
                                    (stand (1 1 1 0))
                                    (precip (2))
                                    (temp (0))
                                    (hail (0 0 0 1))
                                    (years (0 1 2 3))
                                    (damage (1))
                                    (severity (1 2))
                                    (treatment (0 1))
                                    (germ ( 1 2))
                                    (height (1))
                                    (cond (0))
                                    (lodging (0 0 0 1))
                                    (cankers (1))
                                    (color (1))
                                    (fruit (0))
                                    (decay (1))
                                    (mycelium (0 1))
                                    (intern (0))
                                    (sclerotia (0))
                                    (pod (3))
                                    (root (0)))))

;;;
(defvar SOY-D-SPEC (make-instance 'soy-specification
                                  :common-name 'd*
                                  :feature-choice-list
                                  '((time  (0 1 2 3))
                                    (stand (1))
                                    (precip (2))
                                    (temp (0 1))
                                    (hail (0 0 0 1))
                                    (years (0 1 1 2 3 3))
                                    (damage (1))
                                    (severity (1 2))
                                    (treatment (0 1))
                                    (germ (0 1 2))
                                    (height (1))
                                    (cond (1))
                                    (lodging (0))
                                    (cankers (1 2))
                                    (color (2))
                                    (fruit (0))
                                    (decay (0 0 0 1))
                                    (mycelium (0))
                                    (intern (0))
                                    (sclerotia (0))
                                    (pod (3))
                                    (root (1)))))
```

;;;; Bacteria Specifications

```
(defvar BACTERIA-BF-SPEC (make-instance 'bacteria-specification
                                        :genus 'bacteroides
                                        :species 'fragilis
                                        :common-name 'bf
                                        :feature-choice-list
                                        '((location (GI))
                                          (gram (NEG))
                                          (gr-pen (R))
                                          (gr-rif (S))
                                          (gr-kan (R))
                                          (dole (neg))
                                          (esculin (pos))
                                          (bile (e))
                                          (glc (ls))
                                          ;(salicin (neg))
                                          ;(arab (neg))
                                          (rham (neg))
                                          (nf1 (1 2 3 4))
                                          (nf2 ( 1 2 3 4)))))

(defvar BACTERIA-BT-SPEC (make-instance 'bacteria-specification
                                        :genus 'bacteroides
                                        :species 'thetaiotamicron
                                        :common-name 'bt
                                        :feature-choice-list
                                        '((location (GI))
                                          (gram (NEG))
                                          (gr-pen (R))
                                          (gr-rif (S))
                                          (gr-kan (R))
                                          (dole (pos))
                                          (esculin (pos))
                                          (bile (e))
                                          (glc (ls))
                                          ;(salicin (neg pos))
                                          ;(arab (pos))
                                          (rham (pos))
                                          (nf1 (1 2 3 4))
                                          (nf2 ( 1 2 3 4)))))

(defvar BACTERIA-BV-SPEC (make-instance 'bacteria-specification
                                        :genus 'bacteroides
                                        :species 'vulgatus
                                        :common-name 'bv
                                        :feature-choice-list
                                        '((location (GI))
                                          (gram (NEG))
                                          (gr-pen (R))
                                          (gr-rif (S))
                                          (gr-kan (R))
                                          (dole (neg))
                                          (esculin (neg))
                                          (bile (e))
                                          (glc (ls))
                                          ;(salicin (neg pos))
                                          ;(arab (pos))
                                          (rham (pos))
                                          (nf1 (1 2 3 4))
                                          (nf2 ( 1 2 3 4)))))
```

```
(defvar BACTERIA-FM-SPEC (make-instance 'bacteria-specification
                                        :genus 'fusobacterium
                                        :species 'mortiferum
                                        :common-name 'fm
                                        :feature-choice-list
                                        '((location (OR))
                                          (gram (NEG))
                                          (gr-pen (R S))
                                          (gr-rif (R))
                                          (gr-kan (S))
                                          (dole (neg))
                                          (esculin (pos))
                                          (bile (e))
                                          (glc (none))
                                          ;(salicin (neg pos))
                                          ;(arab (neg pos))
                                          (rham (neg pos))
                                          (nf1 (1 2 3 4))
                                          (nf2 ( 1 2 3 4)))))

(defvar BACTERIA-FN-SPEC (make-instance 'bacteria-specification
                                        :genus 'fusobacterium
                                        :species 'necrophorum
                                        :common-name 'fn
                                        :feature-choice-list
                                        '((location (OR))
                                          (gram (NEG))
                                          (gr-pen (S))
                                          (gr-rif (S))
                                          (gr-kan (S))
                                          (dole (pos))
                                          (esculin (neg))
                                          (bile (I))
                                          (glc (none))
                                          ;(salicin (neg))
                                          ;(arab (neg pos))
                                          (rham (neg pos))
                                          (nf1 (1 2 3 4))
                                          (nf2 ( 1 2 3 4)))))

(defvar BACTERIA-FV-SPEC (make-instance 'bacteria-specification
                                        :genus 'fusobacterium
                                        :species 'varium
                                        :common-name 'fv
                                        :feature-choice-list
                                        '((location (OR))
                                          (gram (NEG))
                                          (gr-pen (R S))
                                          (gr-rif (R))
                                          (gr-kan (S))
                                          (dole (pos))
                                          (esculin (neg))
                                          (bile (e))
                                          (glc (none))
                                          ;(salicin (neg))
                                          ;(arab (neg pos))
                                          (rham (neg pos))
                                          (nf1 (1 2 3 4))
                                          (nf2 ( 1 2 3 4)))))
```

;;; Specifications for the simulated two process world.

```
(defvar  GS11-SPEC (make-instance 'species-specification
                                  :genus 'one
                                  :species 'one-one
                                  :common-name 'GS11
                                  :feature-choice-list
                                  '((gf1 (1))
                                    (gf2 (1))
                                    (gf3 (1))
                                    (gf4 (1))
                                    (gf5 (1))
                                    (sf1 (1))
                                    (sf2 (1))
                                    (nf1 (1 2 3 4 5))
                                    (nf2 (1 2 3 4 5))
                                    )))

(defvar  GS12-SPEC (make-instance 'species-specification
                                  :genus 'one
                                  :species 'one-two
                                  :common-name 'GS12
                                  :feature-choice-list
                                  '((gf1 (1))
                                    (gf2 (1))
                                    (gf3 (1))
                                    (gf4 (1))
                                    (gf5 (1))
                                    (sf1 (2))
                                    (sf2 (2))
                                    (nf1 (1 2 3 4 5))
                                    (nf2 (1 2 3 4 5))
                                    )))

(defvar  GS13-SPEC (make-instance 'species-specification
                                  :genus 'one
                                  :species 'one-three
                                  :common-name 'GS13
                                  :feature-choice-list
                                  '((gf1 (1))
                                    (gf2 (1))
                                    (gf3 (1))
                                    (gf4 (1))
                                    (gf5 (1))
                                    (sf1 (3))
                                    (sf2 (3))
                                    (nf1 (1 2 3 4 5))
                                    (nf2 (1 2 3 4 5))
                                    )))
```

```
(defvar  GS21-SPEC (make-instance 'species-specification
                                  :genus 'two
                                  :species 'two-one
                                  :common-name 'GS21
                                  :feature-choice-list
                                  '((gf1 (2))
                                    (gf2 (2))
                                    (gf3 (2))
                                    (gf4 (2))
                                    (gf5 (2))
                                    (sf1 (1))
                                    (sf2 (1))
                                    (nf1 (1 2 3 4 5))
                                    (nf2 (1 2 3 4 5))
                                    )))

(defvar  GS22-SPEC (make-instance 'species-specification
                                  :genus 'two
                                  :species 'two-two
                                  :common-name 'GS22
                                  :feature-choice-list
                                  '((gf1 (2))
                                    (gf2 (2))
                                    (gf3 (2))
                                    (gf4 (2))
                                    (gf5 (2))
                                    (sf1 (2))
                                    (sf2 (2))
                                    (nf1 (1 2 3 4 5))
                                    (nf2 (1 2 3 4 5))
                                    )))

(defvar  GS23-SPEC (make-instance 'species-specification
                                  :genus 'two
                                  :species 'two-three
                                  :common-name 'GS23
                                  :feature-choice-list
                                  '((gf1 (2))
                                    (gf2 (2))
                                    (gf3 (2))
                                    (gf4 (2))
                                    (gf5 (2))
                                    (sf1 (3))
                                    (sf2 (3))
                                    (nf1 (1 2 3 4 5))
                                    (nf2 (1 2 3 4 5))
                                    )))
```

Appendix B

Lambda Tracking

In chapter 6 we described a recursive categorization procedure capable of recovering natural categories in a multiple modal world. We illustrated the technique using the example of anaerobic bacteria; both the genera and species levels of categories were recovered. In this section we provide an alternative mechanism for recovering multiple stable structures within a population. This technique has the desirable property of providing an explicit measure of the degree of structure contained within each separate category.

Recall that for λ near zero, the finest possible categorization — a categorization in which each object is its own category — yields the lowest total categorization uncertainty U. As λ is increased, coarser, less homogeneous categories are preferred. When λ is close to 1.0 the best possible categorization consists of only one category. Thus, we can design an *agglomerative* clustering technique [Duda and Hart, 1973, also chapter 3] which forms new categories by merging previous categories.

The algorithm we use is identical to that introduced in chapter 5 except that λ is no longer constant. We begin by categorizing a population of objects with λ set to some low value. Such a setting causes categories to be continually split, yielding a categorization of many, highly similar categories. Then, as new objects are observed, we slowly increase the value of λ. For each value of λ, the algorithm is permitted to execute until a stable categorization is achieved. As the value of λ increases, categories begin to merge. Finally, as λ approaches 1.0, the categories are merged into a single category. Because we can track the categorization as the value of λ changes, we refer to this algorithm as λ-*tracking*.

To illustrate the behavior of the algorithm, we will use the soybean diseases introduced in chapter 5 and re-presented in table B.1. A population of 20 examples of each species of disease is generated. Because the value of λ changes over time, this technique can only be applied to a fixed population; when the algorithm terminates at λ equal to 1.0, there is only one category and the introduction of a new object would be meaningless.

Normally, agglomerative techniques produce a *dendrogram*, a diagram of reflecting the dynamic change in category structure as the distance between categories is increased (as defined by some metric). For the technique described here, we will display the results of the execution in the form of a λ-space diagram (Figure B.1). For each value of λ a qualitative description of the categorization produced is illustrated. For example, at λ of .55, the categorization produced consists of three categories corresponding to diseases A, B, and C, and several smaller categories each containing members of disease D.

We begin with a λ of .35. This value of λ was found to be sufficiently low to cause the categorization procedure to continually split previous categories. When λ increases to .40, the separate categories containing members of disease B coalesce to form a category corresponding to that disease. Notice that this category remains until λ is raised to a value of .95. We refer to this duration as the λ-stability of the B category. This version of λ-stability is different than that presented in chapter 6 which referred to the stability of the categorization as a whole. In this case, λ-stability permits us to consider the stability of each category individually. In Figure B.1 the four natural categories corresponding to the four diseases display a (relatively) high degree a λ-stability indicating that these categories correspond to natural structure in the population.

Notice, however, that the category {C, D} exhibits the same degree of λ-stability as the D category. Such stability may indicate that there exists a common structure shared by these two diseases that qualifies them as being a natural mode. However, without independent verification from botanists we cannot confirm this hypothesis.

Experimental evaluation indicates that the λ-tracking algorithm is not as powerful a technique for recovering multiple modal levels as is recursive categorization. One explanation for this result is that the algorithm always considers the entire population as a whole, without limiting its attention to

	A *Diaporthe Stem Canker*	B *Charcoal Rot*	D *Rhizoctonia Root Rot*	D *Phytophthora Rot*
time	{3,4,5,6}	{3,4,5,6}	{0,2,3,4}	{0,1,2,3}
stand	0	0	{1,0}	1
precip	2	0	2	2
temp	1	{1,2}	0	{0,1}
hail	0	{0,1}	{0,1}	0
years	{1,2,3}	{0,1,2,3}	{0,1,2,3}	{0,1,2,3}
damage	{0,1}	{2,3}	1	1
severity	{1,2}	1	{1,2}	{1,2}
treatment	{0,1}	{0,1}	{0,1}	{0,1}
germ	{0,1,2}	{0,1,2}	{1,2}	{0,1,2}
height	1	1	1	1
cond	1	1	0	1
lodging	{0,1}	{0,1}	0	0
cankers	3	0	1	{1,2}
color	{0,1}	3	1	2
fruit	1	0	0	0
decay	1	0	1	{0,1}
mycelium	0	0	{0,1}	0
intern	0	2	0	0
sclerotia	0	1	0	0
pod	0	0	3	3
root	0	0	0	1

Table B.1: The property specifications for four species of soybean plant diseases. These data are derived from Stepp [1985].

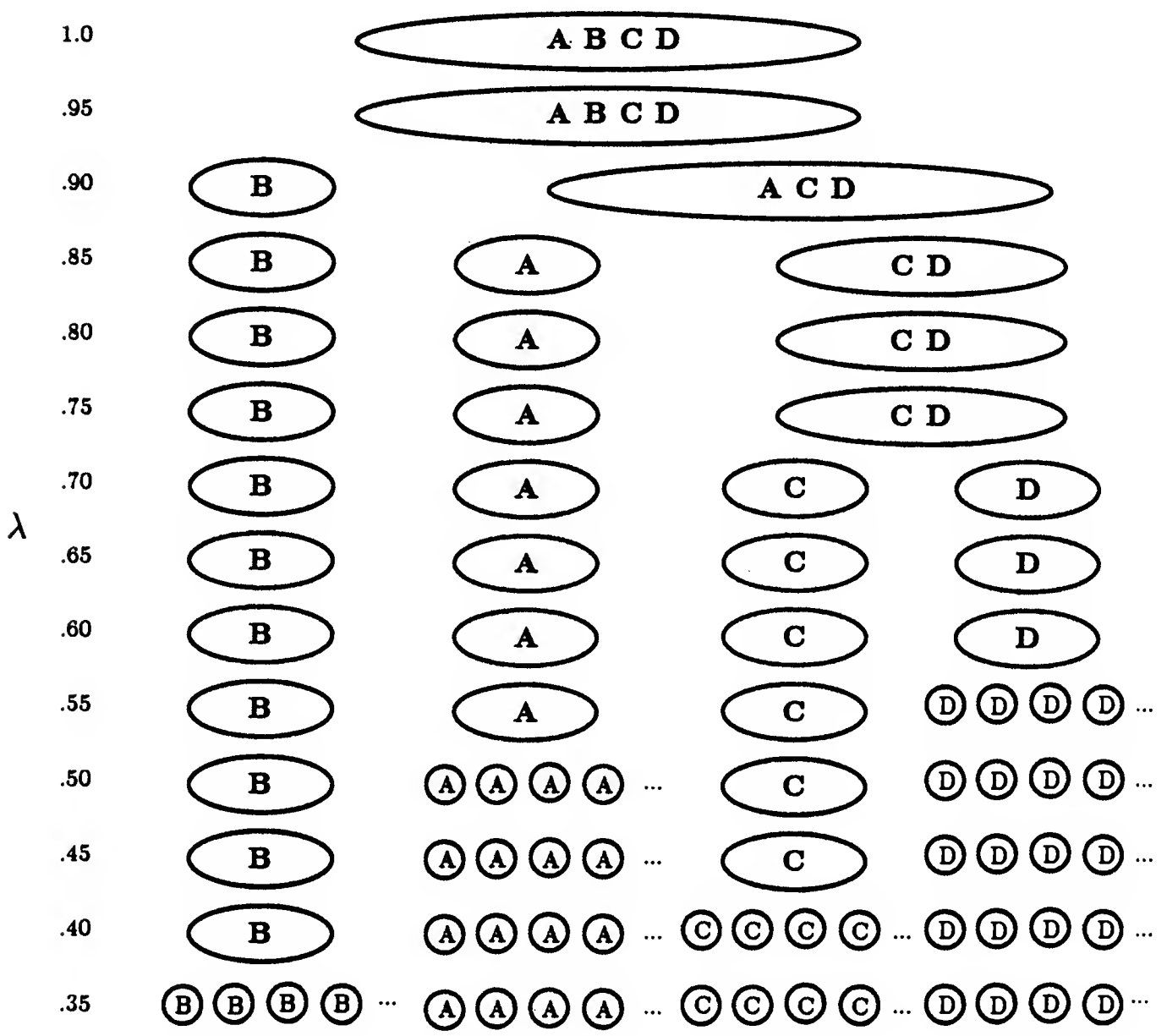


Figure B.1: The λ-space diagram produced by executing the λ-tracking algorithm on a population of soybean diseases. For each λ, a qualitative description of the categorization is illustrated.

finding modes within one particular category. Thus, only small ranges of λ are available for each stable categorization. For example, if there are four stable categorizations, then the maximum λ-stability range for each categorization would be .25. Thus, although λ-tracking allows the assessment of the degree of structure present in each category, it is not a robust mechanism for recovering multiple modal categorizations.